

InfographicVQA

Minesh Mathew¹

Dimosthenis Karatzas³

¹CVIT, IIIT Hyderabad, India

¹IISER Pune, India

Viraj Bagal^{2*}

Rubèn Tito³

Ernest Valveny³

C.V. Jawahar¹

³CVC, Universitat Autònoma de Barcelona, Spain

minesh.mathew@research.iiit.ac.in, viraj.bagal@students.iiserpune.ac.in, rperez@cvc.uab.cat

Abstract

Infographics communicate information using a combination of textual, graphical and visual elements. In this work, we explore automatic understanding of infographic images by using a Visual Question Answering technique. To this end, we present InfographicVQA, a new dataset that comprises a diverse collection of infographics along with question-answer annotations. The questions require methods to jointly reason over the document layout, textual content, graphical elements, and data visualizations. We curate the dataset with emphasis on questions that require elementary reasoning and basic arithmetic skills. For VQA on the dataset, we evaluate two strong baselines based on state-of-the-art Transformer-based, scene text VQA and document understanding models. Poor performance of both the approaches compared to near perfect human performance suggests that VQA on infographics that are designed to communicate information quickly and clearly to human brain, is ideal for benchmarking machine understanding of complex document images. The dataset, code and leaderboard will be made available at docvqa.org

1. Introduction

Infographics are documents created to convey information in a compact manner using a combination of textual and visual cues. The presence of the text, numbers and symbols, along with the semantics that arise from their relative placements, make infographics understanding a challenging problem. True document image understanding in this domain requires methods to jointly reason over the document layout, textual content, graphical elements, data visualisations, color schemes and visual art among others. Motivated by the multimodal nature of infographics, and the human centred design, we propose a Visual Question Answering (VQA) approach to infographics understanding.

VQA received significant attention over the past few years [17, 5, 18, 22, 27, 3]. Several new VQA branches



How many companies have more than 10K delivery workers?

Answer: 2

Evidence: Figure

Answer-source: Non-extractive Operation: Counting Sorting

Who has better coverage in Toronto - Canada post or Amazon?

Answer: canada post

Evidence: Text

Answer-source: Question-span Image-span Operation: none

In which cities did Canada Post get maximum media coverage?

Answer: vancouver, montreal

Evidence: Text Map

Answer-source: Multi-span

Operation: none

Figure 1: Example image from InfographicVQA along with questions and answers. For each question, source of the answer, type of evidence the answer is grounded on, and the discrete operation required to find the answer are shown.

focus on images with text, such as answering questions by looking at text books [28], business documents [36], charts [25, 26, 11] and screenshots of web pages [47]. Still, infographics are unique in their combined use, and purposeful arrangement of visual and textual elements.

In this work, we introduce a new dataset for VQA on infographics, InfographicVQA, comprising 30,035 questions over 5,485 images. An example from our dataset is shown in Figure 1. Questions in the dataset include questions grounded on tables, figures and visualizations as well as questions that require combining multiple cues. Since most infographics contain numerical data, we collect questions

*Work done during an internship at IIIT Hyderabad.

Dataset	Images	Synthetic Images	Template questions	Text type	# Images	# Questions	Answer type
TQA [28]	Science diagrams	✗	✗	MR	1K	26K	MCQ
RecipeQA [54]	Culinary pictures	✗	✓	MR	251K	37K	MCQ
ST-VQA [7]	Natural images	✗	✗	ST	23K	31K	Ex
TextVQA [45]	Natural images	✗	✗	ST	28K	45K	Ex, SAb
OCR-VQA [37]	Book covers	✗	✓	BD	207K	1M	Ex, Y/N
DVQA [25]	Bar charts	✓	✓	BD	300K	3.4M	Ex, Nm, Y/N
FigureQA [26]	Charts - 5 types	✓	✓	BD	120K	1.5M	Y/N
LEAF-QA [11]	Charts - 4 types	✓	✓	BD	250K	2M	Ex, Nm, Y/N
VisualMRC [47]	Webpage screenshots	✗	✗	BD	10K	30K	Ab
DocVQA [36]	Industry documents	✗	✗	Pr, Tw, Hw, BD	12K	50K	Ex
InfographicVQA	Infographics	✗	✗	BD	5.4K	30K	Ex, Nm

Table 1: **Summary of VQA and Multimodal QA datasets where text on the images need to be read to answer questions.** Text type abbreviations are: Machine Readable: MR, Scene Text: ST, Born Digital: BD, Printed: Pr, Handwritten: Hw, and Typewritten: Tw. Answer type abbreviations are: Multiple Choice Question: MCQ, Extractive: Ex, Short abstractive: SAb, Abstractive: Ab, Yes/No: Y/N, and Numerical (answer is numerical and not extracted from image or question; but derived): Nm.

that require elementary reasoning skills such as counting, sorting and arithmetic operations. We believe our dataset is ideal for benchmarking progress of algorithms at the meeting point of vision, language and document understanding.

We adapt a multimodal Transformer [48]-based VQA model called M4C [21] and a layout-aware, BERT [14]-style extractive QA model called LayoutLM [52] for VQA on InfographicVQA. Results using these two strong baselines show that current state-of-the-art (SoTA) models for similar tasks perform poorly on the new dataset. The results also highlight the need to devise better feature extractors for infographics, different from bottom-up features [4] that are typically used for VQA on natural scene images.

2. Related works

Question answering in a multimodal context. Textbook Question Answering (TQA) [28] and RecipeQA [54] are two works addressing Question Answering (QA) in a multimodal context. For TQA, contexts are textbook lessons and for RecipeQA, contexts are recipes containing text and images. Contrary to InfographicVQA and other datasets mentioned below, text in these two datasets are not embedded on the images, but provided in machine readable form, as a separate input.

ST-VQA [7] and TextVQA [45] datasets extend VQA over natural images to a new direction where understanding scene text on the images is necessary to answer the questions. While these datasets comprise images captured in the wild with sparse text content, InfographicVQA has born-digital images with an order of magnitude more text tokens per image, richer in layout and in the interplay between textual and visual elements. OCR-VQA [37] introduces a task similar to ST-VQA and TextVQA, but solely on images of book covers. Template questions are generated from book metadata such as author name, title and other information. Consequently, questions in this dataset rely less on

visual information present in the images. DVQA [25], FigureQA [26], and LEAF-QA [11] datasets deal with VQA on charts. All the three datasets have chart images rendered using chart plotting libraries and template questions.

DocVQA [36] is a VQA dataset that comprises document images of industry/business documents, and questions that require understanding document elements such as text passages, forms, and tables. Similar to ST-VQA, DocVQA is an extractive VQA task where answers can always be extracted verbatim from the text on the images. VisualMRC [47] on the other hand is an abstractive VQA (answers cannot be directly extracted from text in the images or questions) benchmark where images are screenshots of web pages. Compared to VisualMRC, InfographicVQA is an extractive VQA task (answers are extracted as ‘span’(s) of the question or text present in the given image), except for questions that require certain discrete operations resulting in numerical non-extractive answers. (see subsection 3.2). In Table 1 we present a high level summary of the QA/VQA datasets related to ours.

Multimodal transformer for Vision-Language tasks.

Following the success of BERT [14]-like models for Natural Language Processing (NLP) tasks, there have been multiple works extending it to the Vision-Language space. Models like VL-BERT [46], VisualBERT [31], and UNITER [12] show that combined pretraining of BERT-like architectures on vision and language inputs achieve SoTA performances on various downstream tasks including VQA on natural images. For VQA on images with scene text, M4C and TAP [55] use a multimodal transformer block to fuse embeddings of question, scene text tokens, and objects detected from an image.

The success of transformer-based models for text understanding inspired the use of similar models for document image understanding. LayoutLM and LAMBERT [16] incorporate layout information to the BERT architecture by

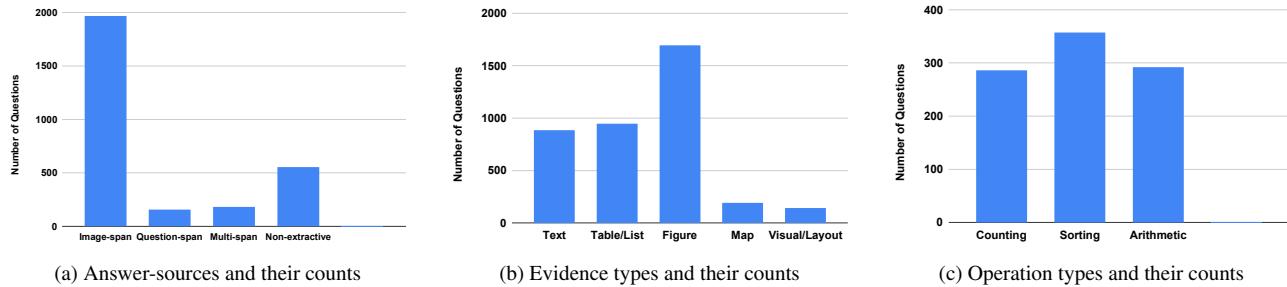


Figure 2: Count of questions in validation set by their Answer-source, (2a), Evidence required to answer (2b) and the discrete Operation performed to find the answer (2c).

using embeddings of the 2D positions of the text tokens in the image. One of the strong baselines we use in this work is based on the LayoutLM model. Concurrent to this work, there have been multiple works published on arXiv that deal with joint understanding of text, image and layout in document images. Models such as LayoutLMv2 [53], TILT [39], DocFormer [6] and StrucText [32] build on transformer-based architectures and leverage large-scale pretraining on unlabelled data, using pretraining objectives specifically designed for document understanding.

Infographics understanding. Bylinskii *et al.* [10] and Madan *et al.* [35] looked at generating textual and visual tags from infographics. In another work, Landman uses an existing text summarization model to generate captions for infographics [30]. This model uses only text recognized from infographics to generate the captions and layout/visual information is not considered. These three works use Visually29K dataset comprising images from a single website. MASSVIS [9] is a collection of infographics created to study infographics from a cognitive perspective. As observed by Lu *et al.* [33], MASSVIS is a specialized collection focusing on illustration of scientific procedures and statistical charts, therefore not representative of general infographics.

To summarize, existing datasets containing infographics are either specialized collections or infographics collected from a single source. In contrast, the InfographicVQA dataset comprises infographics drawn from thousands of different sources, with diverse layouts and designs, and without any topic specialization.

3. InfographicVQA

A brief description of the data collection and detailed analysis of the data is presented here. For more details on data collection, annotation process and annotation tool, refer to Section A in the supplementary material.

3.1. Collecting images and question-answer pairs

Infographics in the dataset were downloaded from the Internet for the search query “infographics”. The downloaded

images are cleaned for removal of duplicates before adding them to the annotation tool. Unlike crowd sourced annotation, InfographicVQA was annotated by a small number of annotators using an internal annotation tool. The annotation process involved two stages. In the first stage, workers were required to add question-answer pairs on an image shown. Similarly to SQuAD dataset [40] annotation, in order to make the evaluation more robust, we collect an additional answer for each question in the validation and test split by sending those questions through a second stage of annotation. In this stage an image along with questions asked on it in the first stage are shown to a worker. Workers were instructed to enter answers for the questions shown or flag a question if it is unanswerable.

3.2. Question-answer types: answer-source, evidence and operation

In the second stage of annotation, in addition to answering questions collected in the first stage, we instructed the workers to add question-answer types (QA types). QA types are a set of category labels assigned to each question-answer pair. DocVQA and VisualMRC have QA types that indicate the kind of document object (table, form, title etc.) a question is based on. DROP [15] dataset for reading comprehension define answer types such as Question span and Passage span and categorize questions by the kind of discrete operations (count, add etc.) required to find the answer. In InfographicVQA we collect QA types under three categories — Answer-source, Evidence and Operation.

There are four types of Answer-source — Image-span, Question-span, Multi-span and Non-extractive. Akin to the definition of ‘span’ in SQuAD [40] and DocVQA, an answer is considered Image-span if it corresponds to a single span (a sequence of text tokens) of text, extracted verbatim, in the reading order, from text present in the image. Similarly when the answer is a span from the question it is labelled as Question-span. In Figure 1 the answer to the second question is both an Image-span and a Question-span since ‘canada post’ appears as a single sequence of contiguous tokens (or a ‘span’) both in the question and in the

image. A Multi-span answer is composed of multiple spans of text from the image. Like in the case of DROP dataset annotation, we instructed our workers to enter Multi-span answers by separating each individual span by a comma and a white space. For example in [Figure 1](#) for the last question the answer is names of two cities. And the names do not appear in a contiguous sequence of text. Hence it is a Multi-span answer. For a Multi-span answer, any order of the individual spans is a valid answer. In case of the above example, both “Vancouver, Montreal” and “Montreal, Vancouver” are valid answers. Since such answers are unordered lists, at evaluation time we consider all permutations of the list as valid answers for the question. The ‘Non-extractive’ type is assigned when the answer is not an extracted one. While collecting question-answer pairs, Non-extractive questions were allowed only if the answer is a numerical value. Inclusion of Question-span, Multi-span and numerical Non-extractive answers in InfographicVQA is inspired by similar setting in the DROP dataset. We see this as a natural next step in VQA involving text, different from the purely extractive QA setting in datasets like DocVQA and ST-VQA, and abstractive question answering in VisualMRC where automated evaluation is difficult. By allowing only numerical answers in Non-extractive case, we make sure that such answers are short and unique, giving no room for variability. Near perfect human performance while using automatic evaluation metrics ([Table 4](#)) validates that answers in InfographicVQA are unique with minimal variability when answered by different individuals.

The Evidence type indicates the kind of evidence behind the answer. Types of evidence are Text, Figure, Table/List, Map and Visual/Layout. For example Map is used if the question is based on data shown on a geographical map. Visual/Layout type is added when evidence is based on the visual or layout aspect of the image. For example, questions such as “What is the color of the hat - brown or black?” or “What is written at the top left corner” fall in this category. Sometimes it is difficult to discern evidence for a question-answer pair. For example, for the first question in [Figure 1](#), although the evidence type assigned by the worker is ‘Figure’, it could even be ‘Table/List’ since the visualization looks like a table. The operation type captures the kind of discrete operation(s) required to arrive at an answer — Counting, Arithmetic or Sorting.

In [Figure 2](#) we show the distribution of questions in the validation split based on Answer-source, Evidence and Operation. As evident from [Figure 1](#), a question can have multiple types of answer source, evidence, or operation and many questions do not require any of the specified discrete operations to find the answer. For these reasons, counts in plots shown in [Figure 2](#) do not add up to 100%.

Dataset	Questions		Answers		Avg. tokens per image
	%Unique	Avg. len	%Unique	Avg. len	
ST-VQA	84.84	8.80	65.63	1.56	7.52
TextVQA	80.36	8.12	51.74	1.51	12.17
VisualMRC	96.26	10.55	91.82	9.55	151.46
DocVQA	72.34	9.49	64.29	2.43	182.75
InfographicVQA	99.11	11.54	48.84	1.60	217.89

Table 2: Statistics of questions, answers and OCR tokens in InfographicVQA and other similar VQA datasets.

3.3. Summary, statistics, and analysis

InfographicVQA dataset has 30,035 questions and 5,485 images in total. These images are from 2,594 distinct web domains. The data is split randomly to 23,946 questions and 4,406 images in train, 2,801 questions and 500 images in validation, and 3,288 questions and 579 images in test splits. We show basic statistics of questions, answers and OCR tokens in InfographicVQA and other similar datasets in [Table 2](#).

Questions. [Table 2](#) shows that InfographicVQA has highest percentage of unique questions and highest average question length compared to other similar datasets. [Figure 3](#) shows a sunburst of the common questions in the dataset. There are a good number of questions asking for “How many...” or percentages. This is expected since infographics carry a lot of numerical data.

Answers. It can be seen in [Figure 4](#) that the most common answers are numbers. This is the reason why InfographicVQA has a smaller number of unique answers and smaller average answer lengths.

Embedded text. The number of average text tokens in InfographicVQA images is 217.89. Word cloud of Optical

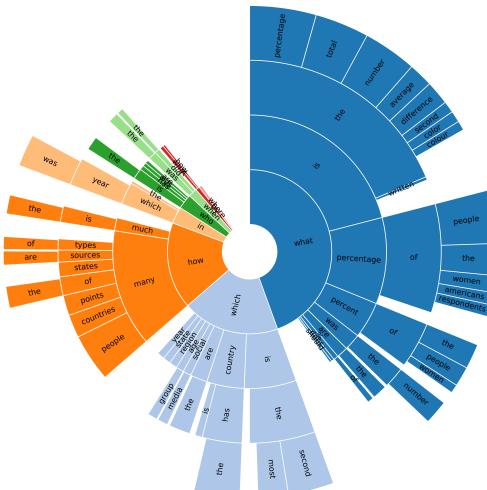


Figure 3: Staring 4-grams of common questions in InfographicVQA.



Figure 4: Word cloud of answers (left) and word cloud of words recognized from the infographics (right).

Character Recognition (OCR) tokens spotted on the images is shown in the wordcloud on the right in Figure 4. Common answers and common text tokens in images are similar.

4. Baselines

In this section, we describe the baselines we evaluated on the InfographicVQA. These include heuristic baselines and upper bounds, and SoTA models for VQA and document understanding.

4.1. Heuristic baselines and upper bounds

Heuristic baselines and upper bounds we evaluate are similar to the ones evaluated in other VQA benchmarks like TextVQA and DocVQA.

Heuristic Baselines. We evaluate performance when answer predicted for each question is i) **Random answer** from the train split, ii) **Majority answer** from the train split and iii) **Random OCR token** from the image on which question is asked.

Upper bounds. We evaluate the performance upper bound on predicting the correct answer if the answer is present in a vocabulary of most common answers in train split. This upper bound is called **Vocab UB**. Following DocVQA, to assess the percentage of questions for which answers can be found from the given OCR transcriptions, we compute **OCR UB** which measures the upper bound on performance if we always predict the correct answer, provided answer is a sub sequence of the serialized OCR transcription of the given infographic. We serialize OCR tokens in the natural reading order , i.e., from top-left to bottom-right. **Vocab+OCR UB** is the percentage of questions which are either in Vocab UB or OCR UB.

4.2. M4C

M4C uses a Transformer stack to fuse representations of a question, OCR tokens, and image. Answers are predicted using an iterative, auto-regressive decoder decoding one word at a time, either from a fixed vocabulary or from the OCR tokens spotted on the image. Original M4C uses Region of Interest (ROI) pooled features from Box head of a Faster-RCNN [42] as the bottom-up visual features. Visual features are used for both the objects detected on the image and the OCR tokens spotted on the image.

4.3. LayoutLM

LayoutLM [52] extends BERT by incorporating layout information into the original BERT model. LayoutLM pre-trained on millions of document images has proven to be effective for multiple document image understanding tasks. We adapt LayoutLM for InfographicVQA by changing the input to suit a multimodal setting and using an output head for SQuAD-style span prediction. Since we are using SQuAD-style span prediction at the output, this model can only handle questions whose Answer-source is Image-span. Nearly 70% of questions in validation and test split are of this type. Extending this model to include questions with other Answer-sources is an interesting direction for future work.

4.3.1 Model overview

A schematic of the LayoutLM-based model which we use for InfographicVQA is shown in Figure 5. Input sequence of tokens to the model are composed of the tokens in question and the tokens spotted on the image by an OCR. The sequence starts with a special [CLS] token, followed by the question tokens and the OCR tokens. The sequence ends with a special [SEP] token. Question and OCR tokens are also separated by a [SEP] token. All the tokens in the input sequence are represented by a corresponding embedding which in turn is the sum of i) a token embedding, ii) a segment embedding, iii) a 1D position embedding, iv) Four 2D position embeddings, and v) a visual embedding.

Following the original setting in BERT, for **token embedding**, we use WordPiece embeddings [51] with a 30,000 size vocabulary. These embeddings are of size 768. A **segment embedding** differentiates different segments in the input. For example when the input sequence is formed by tokens from question and OCR tokens, question tokens and OCR tokens are given a segment id of 0 and 1 respectively. **1D position embedding** is used to indicate the order of a token within the sequence. For OCR tokens we use the default reading order by serializing the tokens in top-left to bottom-right order.

For 2D position embedding, we follow the same process as original LayoutLM. Given the bounding box coordinates (x_1, y_1, x_2, y_2) of an OCR token, we embed all the 4 coordinate values using 4 separate embedding layers. x_1 and x_2 share same embedding table and y_1 and y_2 share another common embedding table. The coordinate values are normalized to lie in the range 0–1000 before embedding. For [CLS] token we use 2D embedding corresponding to $(0, 0, 1000, 1000)$. For question tokens all the four 2D position embeddings used are 0s.

Unlike in original LayoutLM where visual features are fused after getting the attended embeddings from the Transformer block, we fuse the visual features early with the text

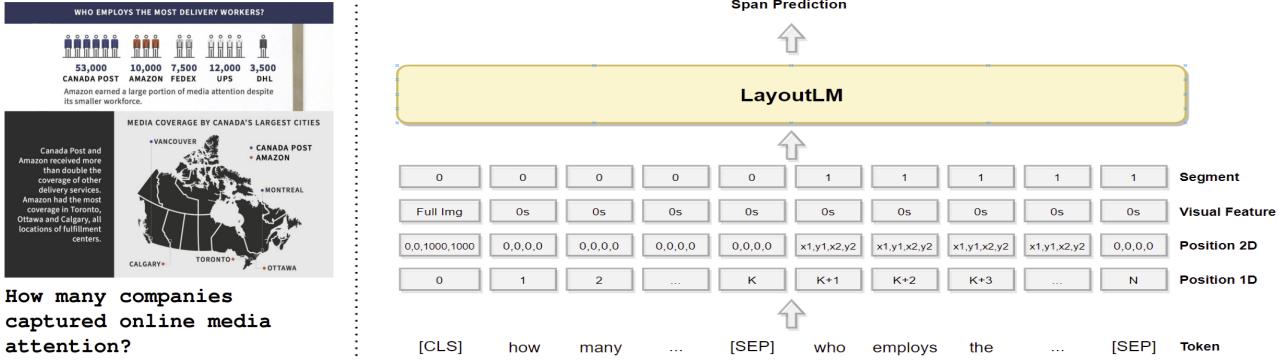


Figure 5: Overview of our LayoutLM based model for predicting answer spans. Textual, visual and layout modalities are embedded and mapped to the same space. Then, they are added and passed as input to a stack of Transformer layers.

input. Similar to M4C, for each OCR token, we use ROI pooled feature from the Box head of a pretrained object detection style model. This feature is mapped to same size as other embeddings using a linear projection layer. For [CLS] and other special tokens we add visual feature corresponding to an ROI covering the entire image, named as “Full Img” in Figure 5.

4.3.2 Training procedure

Similar to the BERT and original LayoutLM, we train the model in two stages.

Pretraining: Following original LayoutLM, we use Masked Visual-Language Model (MVLML) task for pre-training with a masking probability of 0.15. Whenever masking, we replace each token with the [MASK] token 80% of the time, with a random token 10% of the time and keep it unchanged 10% of the time.

Finetuning: For finetuning, similar to BERT QA model for SQuAD benchmark, the output head used in pretraining is replaced by a span prediction head that predict start and end token positions of the answer span.

5. Experiments and results

In this section, we describe the experiments we conducted on the new InfographicVQA dataset and present the results and analysis.

5.1. Experimental setup

Evaluation metrics. For evaluating VQA performance on InfographicVQA, we use Average Normalized Levenshtein Similarity (ANLS) and Accuracy metrics. The evaluation setup is exactly same as evaluation in DocVQA.

OCR transcription. Text transcriptions and bounding boxes for text tokens in the images are obtained using Texttract OCR [1].

Human performance For evaluating human performance, all questions in the test split of the dataset are answered

with the help of two volunteers (each question answered by a single volunteer).

Vocabulary of most common answers. For Vocab UB and heuristics involving a vocabulary, we use a vocabulary of 5,000 most common answers in the train split.

ROI Features. For our experiments using M4C and LayoutLM models, visual features of different bounding regions from the images are used. To this end, we use two pretrained object detection models — a Faster-RCNN [42] trained on Visual Genome [29] and a Mask-RCNN [19] trained on document images in PubLayNet [56] for Document Layout Analysis (DLA). We refer to these detectors as VG and DLA respectively in our further discussion. The FasterRCNN model we use is same as the one used for M4C. We use the implementation in MMF framework [43]. The DLA detector we use is from a publicly available Detectron2 [50]-based implementation [20]. Features from the last or second last Fully Connected (FC) layer are used as visual features in M4C and LayoutLM model. In VG and DLA, these features are of size 2048 and 1024 respectively.

In Table 3 we give a summary of the detections using the two detectors on TextVQA, DocVQA and InfographicVQA. With DLA, we notice that many of its detections, especially when there is only one detection per image is a

Detector	TextVQA		DocVQA		InfographicVQA	
	Avg.	<2 det.(%)	Avg.	<2 det.(%)	Avg.	<2 det.(%)
VG	28.8	0.0	4.1	43.9	7.4	23.9
DLA	1.0	97.9	4.7	0.0	2.9	43.4

Table 3: Statistics of object detections using two detectors – VG and DLA. DLA is trained for document layout analysis and VG is an object detection model trained on Visual Genome. Avg. shows average number of detections per image. ‘<2 det.(%)’ is the percentage of images on which number of detected objects is less than 2. Results show that DLA and VG which are suitable for document images and natural images respectively, detect much lesser object instances on infographics.

Baseline	ANLS		Accuracy(%)	
	val	test.	val	test
Human performance	-	0.980	-	95.70
Random answer	0.006	0.005	0.00	0.00
Random OCR token	0.011	0.014	0.29	0.49
Majority answer	0.041	0.035	2.21	1.73
Vocab UB	-	-	53.16	51.34
OCR UB	-	-	53.95	56.96
Vocab + OCR UB	-	-	76.71	77.4

Table 4: Results of heuristics and upper bounds. Heuristics yield near zero results. More than 75% of the questions have their answer present either in a fixed vocabulary or as an Image-span of the OCR tokens serialized in default reading order.

box covering the entire image.

Experimental setting for M4C. We use the official implementation of the model [43]. The training parameters and other implementation details are same as the ones used in the original paper. As done in original M4C, fixed vocabulary used with the model is created from 5,000 most common words among words from answers in the train split.

Experimental setting for LayoutLM. The model is implemented in Pytorch [38]. In all our experiments we start from a pretrained checkpoint of LayoutLM model made available by the authors in Huggingface’s Transformers model zoo [49, 2]. The newly introduced linear projection layer which maps the ROI pooled features to the common embedding size of 768, is initialized from scratch. The features are from the last FC layer of the Box head of DLA or VG. To continue pretraining using in-domain data, we use four samples in one batch and Adam optimizer with learning rate $2e - 5$. For finetuning, we use a batch size of 8 and Adam optimizer with learning rate $1e - 5$. For in-domain pretraining and finetuning no additional data other than train split of InfographicVQA is used. To map answers in InfographicVQA train split to SQuAD [40]-style spans, we follow the same approach used by Mathew *et al.* in DocVQA. We take the first subsequence match of an answer in the serialized transcription as the corresponding span. This way we found approximate spans for 52% of questions in the train split. Rest of the questions are not used for finetuning the model.

5.2. Results

Results of heuristic baselines, upper bounds, and human performance is shown in Table 4. Human performance is comparable to the human performance on DocVQA. As given by the Vocab + OCR UB, more than 3/4 of questions have their answers present as a span of the OCR tokens serialized in the default reading order or in a vocabulary of most common answers in the train split.

We show results using M4C model in Table 5. In contrast to the original setting for which finetuning of visual features and features of detected objects are used, a setting that uses

Visual Feature	Finetune detector	Object& Count	# OCR tokens	ANLS		Accuracy(%)	
				val	test	val	test
VG	✓	Obj. (100)	50	0.107	0.119	4.81	4.87
VG	✓	Obj. (20)	50	0.111	0.122	4.82	4.87
VG	✗	Obj. (20)	50	0.125	0.127	4.89	4.89
VG	✗	Obj. (20)	300	0.128	0.134	4.90	5.08
VG	✗	None	300	0.136	0.143	5.86	6.58
VG	✗	Full Img	300	0.142	0.147	5.93	6.64
DLA	✗	Obj. (20)	50	0.110	0.130	4.86	5.02
DLA	✗	Obj. (20)	300	0.132	0.144	5.95	6.50
DLA	✗	None	300	0.140	0.142	5.90	6.39
DLA	✗	Full Img	300	0.138	0.140	5.97	6.42

Table 5: Performance of different variants of the M4C model. The original M4C setting is the one shown in the first row. ‘Finetune detector’ denotes the case when features from penultimate FC layer is used and last FC layer is finetuned along with the M4C model. This is the default setting in M4C. In our experiments, we get better results without finetuning. ‘Obj. (100)’ is the case when features from up to 100 objects (bottom-up features) are used. We experiment with 20 objects per image and the results did not change much. Using no object (‘None’) and feature from only one object—a box covering the entire image (‘Full Img’)—yield better results than the case where bottom-up objects are used.

no finetuning and only a single visual feature corresponding to ROI covering the entire image, yields the best result.

Results of the LayoutLM based model are shown in Table 6. In-domain pretraining, using text from question, and OCR tokens helps the model significantly. This is inline with observation by Singh *et al.* that pretraining on data similar to the data for downstream task is highly beneficial in visio-linguistic pretraining [44]. The model that uses Full Img feature from DLA, added to the CLS performs the best on validation set. And on test set, a model which does not use any visual feature performs the best.

From Table 6, it is evident that models which use visual

Full Img to	Visual feature	Continue pretrain.	OCR visual	ANLS		Accuracy (%)	
				val	test	val	test
-	-	✗	✗	0.212	0.225	13.40	15.32
-	-	✓	✗	0.250	0.272	18.14	19.74
CLS	DLA	✓	✗	0.256	0.261	18.56	19.16
All	DLA	✓	✗	0.248	0.266	17.82	18.77
Non-OCR	DLA	✓	✓	0.245	0.263	17.21	18.37
CLS	VG	✓	✗	0.229	0.235	16.47	16.51
All	VG	✓	✗	0.109	0.106	5.43	4.96
Non-OCR	VG	✓	✓	0.042	0.037	1.75	1.28

Table 6: Performance of LayoutLM with different input settings. Row 1 and 2 show LayoutLM’s performance with and without in-domain pretraining. ‘Visual Feature’ column specify the kind of detector used for visual feature. ‘OCR visual’ indicate whether visual features of the OCR tokens are used or not. ‘CLS’, ‘All’ and ‘Non-OCR’ in ‘Full Img to’ column represent Full Img feature added only to CLS token, all tokens and all non OCR tokens respectively.

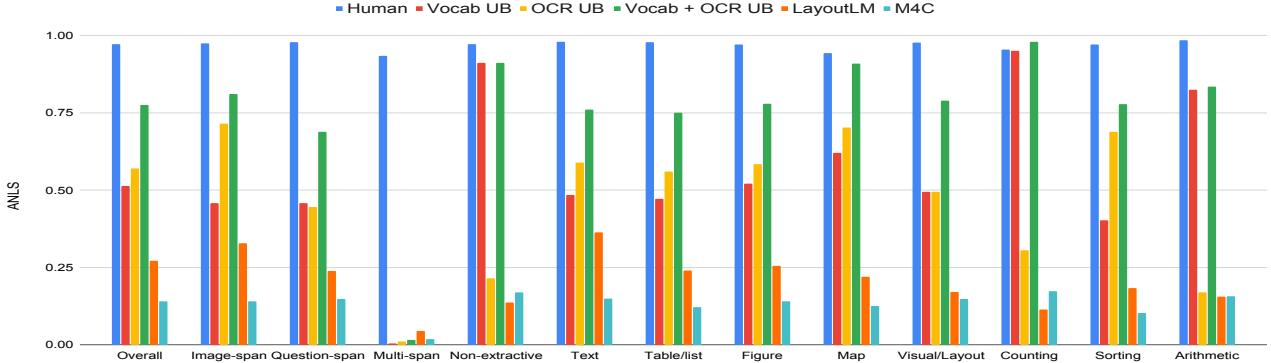
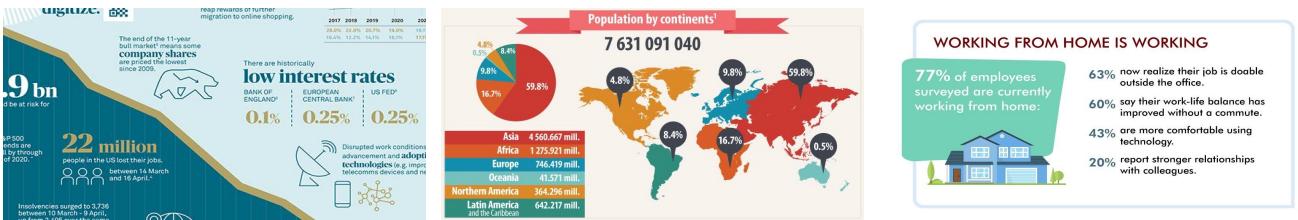


Figure 6: Performance of baselines and upper bounds for different QA types.



What is the interest rates of European Central Bank and US FED?

LayoutLM: 0.25% M4C: 0.1%
Human: 0.25% GT: 0.25%

Which is the least populated continent in the world?

LayoutLM: EU M4C: Oceania
Human: Oceania GT: Oceania

What percentage of workers are not working from home?

LayoutLM: 77% M4C: 66%
Human: 23% GT: 23%

Figure 7: **Qualitative Results** For the left most question, evidence is Table/List and the LayoutLM model gets the prediction right. In case of the second question where evidence is a Table/List and Sorting is involved, M4C gets the answer correct. In case of the last question which requires subtraction of 77 from 100 neither M4C, nor LayoutLM gets the answer correct. (Since original images are much bigger we only show crops of the infographics which are relevant to the question). More qualitative examples showing images in original size are added in the supplementary.

features of OCR tokens do not give better results. This implies that token embeddings of the relatively noise free OCR tokens are good enough and the additional information from visual features of the tokens contribute little to the performance.

Most of the recent models that employ visio-linguistic pretraining of BERT-like architectures [31, 46] incorporate bottom-up visual features—features of objects detected on the images—into the model as visual tokens. We follow approach in VisualBERT [31] where visual tokens are concatenated after the input stream of text tokens. Each visual token is represented by a dummy text token [OBJ], a separate segment, 1D and 2D positions and the ROI pooled visual feature of the object’s region. But in our experiments, the addition of visual tokens did not give us results any better than the model without visual tokens. Hence we do not show this setting in illustration of our model architecture or in the results table. We believe the visual tokens we use impart little information since the object detectors we use—a detector trained for detecting objects on natural scene images and another for document layout analysis—are not

suitable for infographics. This is quite evident in Table 3. Both the detectors detect only a few instances of objects on infographics.

In Figure 6 performance of our trained baselines are compared against the upper bounds and human performance, for different Answer-sources, Evidence and Operation types in test split of the dataset. The M4C and LayoutLM models used for this plot are the variants which give best ANLS on the test data. Finally a few qualitative results from our experiments are shown in Figure 7.

6. Conclusion

We introduce the InfographicVQA dataset and associated task of VQA on infographics. Results using the baseline models suggest that existing models designed for multimodal QA or VQA perform poorly on the new dataset. We believe our work will inspire research towards understanding images with complex interplay of layout, graphical elements and embedded text.

A. Data collection

This section provides details on how we collected infographics from internet, cleaned it and the way we collect questions and answers on these images.

A.1. Collecting images and de-duping

Images in the dataset are sourced from internet. Initially we downloaded more than 10K images for the search query “infographics” using Google and Bing image search engines. The downloaded images were first de-duped using a Perceptual Hashing approach implemented in *imagededup* library [23] this removed nearly 2000 duplicate images. The first round of de-duplication helped to reduce the number of images for the second round of deduplication that involved use of a commercial OCR. In this round, we compared the images using Jaccard similarity of the text tokens spotted on the images using the Amazon Textract OCR [1]. After two rounds of de-duplication, around 7K images were left. These were added to the annotation system for question-answer annotation.

A.2. Question-answer annotation

Here we present details of the annotation such as selection of workers, annotation process and the annotation tool.

Annotation scheme and selection of workers.

Initially we had hosted a pilot annotation on a crowdsourcing platform for collecting question-answer pairs on infographics. But more than 40% of the question-answer pairs added during the pilot annotation were noisy. We realized that some of the requirements were not easy to understand from written instructions. For example the kind of question-answers which are allowed—only questions whose answer sources are one of the four types—is better understood when explained using examples. Consequently we decided to use an internal web-based tool for the annotation and hired workers with whom we could interact closely.

To select the workers, we reached out to individuals looking for annotation-type jobs through mailing lists and other online groups. Interested applicants were invited to join a 90 minute webinar explaining the process and all the requirements. During the webinar we explained them each of the instructions with many examples for accepted type of questions and question-answer types. Following the webinar, the applicants were asked to take an online quiz to assess how well they understood the process and the policies. Based on the quiz scores we selected 13 of them for the annotation. The selected workers were called for another round of webinar where we discussed the answer key for the quiz and clarified their doubts. The workers were added to an online forum so that they could post their

queries related to the annotation in the forum. They were encouraged to post questions with screenshots whenever in doubt. They would keep the particular image in pending and move on to other images in the queue, until one of the authors give a reply to the question they raised. This way we could reduce annotation errors drastically. [Figure A.1](#) shows a screenshot of our web-based annotation tool that shows the interface from which a worker picks the next image for annotation, while having a few documents in pending.

Annotation tool

Annotation of InfographicVQA was organized in two stages. In the first stage, questions-answer pairs were collected on the infographics. Workers were allowed to reject an image if it is not suitable. See [Table A.1](#) for instructions on when to reject an image. After collecting more than 30K questions and their answers, we stopped the first round of annotation, as we were aiming for a dataset with 30K questions in total. We split this data into train, validation and test splits so that the splits roughly have 80, 10 and 10 percentage of the total questions respectively. [Figure A.2](#) shows a screenshot from the stage 1 of the annotation.

Inspired by the SQuAD dataset annotation, we include a second stage in the annotation process to collect additional answers for questions in the validation and test split. Hence only images from validation and test splits were sent through this stage. In this stage, a worker was shown an image and were asked to answer the questions that had been added on the image in the first stage (answers entered in the first stage were not shown). Finally, we retain all the unique answers (i.e., unique strings after converting all answers to lower case) entered for a question. Hence a question can have more than 1 valid answer. We made sure that second stage was done by a worker different from the one who collected questions and answers on the same image in the first stage. The workers were also allowed mark a question as “can’t answer” if they are not able to find an answer for the question based on the information present in the image. Around 1.9% questions were marked so and those were removed from the dataset. As mentioned in subsection 3.2 in the main paper, in the second stage, workers were also required to assign question-answer types based on answer source, evidence and operation for each question. [Figure A.3](#) shows a screenshot from stage 2 of the annotation.

Written instructions for both stages of the annotation that we shared with the workers are given in [Table A.1](#).

A.3. Data release

The dataset will be made publicly available under terms set by our data protection team. The question-answer annotations will be made available under CC-BY license. For

research and educational uses, the images in the dataset will be available for download as single zip file. For other uses, a list of original URLs for all the images in the dataset will be provided.

B. Additional Statistics and Analysis of InfographicVQA dataset

In this section we provide additional statistics and analysis of the InfographicVQA dataset. This section extends Section 3 in the main paper.

To analyze the topics covered by images in InfographicVQA, we used the Latent Dirichlet Allocation (LDA) [8] model. We used the LDA implementation in Gensim library [41]. Since our dataset comprises images, the text recognized from the images using an OCR are used for the topic modelling. Table B.1 shows that images in InfographicVQA dataset cover a wide range of topics such as energy, war, health and social media. In Figure B.1 we show a visualization of the top 20 topics in the dataset, visualized using the pyLDAvis tool [34].

In Figure B.2, we plot the distribution of question lengths for questions in InfographicVQA dataset and other similar datasets — TextVQA [45], ST-VQA [7], DocVQA [36] and VisualMRC [47]. Similar plots for answer lengths and number of OCR tokens are shown in Figure B.3 and Figure B.4 respectively.

Top-15 questions in InfographicVQA dataset based on occurrence frequency are shown in Figure B.5. Top-15 answers and top-15 non-numeric answers are shown in Figure B.6 and Figure B.7 respectively.

For TextVQA dataset, for all statistics we use only the publicly available data splits. For statistics and analysis involving OCR tokens, for InfographicVQA we use OCR tokens spotted by Amazon Textract OCR (we will be making these OCR results publicly available along with our dataset). For TextVQA and ST-VQA we use OCR tokens provided as part of data made available in MMF framework [43]. For DocVQA and VisualMRC we use OCR recognition results made available as part of the official data releases.

C. More details on experiments

C.1. Evaluation

Since more than 70% of the answers in the dataset are taken verbatim from the text present in the image, we decided to keep the evaluation protocol same as the one used for ST-VQA [7] and DocVQA [36] benchmarks. These benchmarks where answers are compulsorily extracted from the text on the images, authors propose to use Average Normalized Levenshtein Similarity (ANLS) as the primary evaluation metric. The metric was originally introduced for evaluating VQA on ST-VQA. As the authors

of ST-VQA state, ANLS “responds softly to answer mismatches due to OCR imperfections”.

The below definition for ANLS is taken from ST-VQA paper.

In Equation C.1 N is the total number of questions and M the number of GT answers per question. a_{ij} are the ground truth answers where $i = \{0, \dots, N\}$, and $j = \{0, \dots, M\}$, and o_{q_i} the predicted answer for the i^{th} question q_i . Then, the final score is defined as:

$$\text{ANLS} = \frac{1}{N} \sum_{i=0}^N \left(\max_j s(a_{ij}, o_{q_i}) \right) \quad (\text{C.1})$$

$$s(a_{ij}, o_{q_i}) = \begin{cases} (1 - NL(a_{ij}, o_{q_i})) & \text{if } NL(a_{ij}, o_{q_i}) < \tau \\ 0 & \text{if } NL(a_{ij}, o_{q_i}) \geq \tau \end{cases}$$

where $NL(a_{ij}, o_{q_i})$ is the Normalized Levenshtein distance between the lower-cased strings a_{ij} and o_{q_i} (notice that the normalized Levenshtein distance is a value between 0 and 1). We then define a threshold $\tau = 0.5$ to filter NL values larger than the threshold. The intuition behind the threshold is that if an output has a normalized edit distance of more than 0.5 to an answer, it is highly unlikely that the answer mismatch is due to OCR error.

In addition to ANLS, we also evaluate the performance in terms of Accuracy which is the percentage of questions for which the predicted answer match exactly with at least one of the ground truth answers. Note that for both the ANLS and Accuracy computation, the ground truth answers and the predicted answers are converted to lowercase.

C.2. Experimental setup for M4C

This section provides additional details about the experimental setup we used for the M4C [21] model. We used the official implementation available as part of the MMF multimodal learning framework [43].

We trained our models on 4, NVIDIA RTX 2080Ti GPUs. The maximum number of decoding steps used for the iterative answer prediction module is 12. The multimodal transformer block had 4 transformer layers, with 12 attention heads. The dropout ratio for the transformer block was 0.1. We used Adam optimizer. The batch size was 128 and we trained the models for 24,000 iterations. We used a base learning rate of $1e - 04$, a warm-up learning factor of 0.2 and 2,000 warm-up iterations. We used a learning rate decay of 0.1. Gradients were clipped when L^2 norm exceeds 0.2.

C.3. Experimental setup for LayoutLM

Preparing QA data for finetuning LayoutLM. We fine-tune LayoutLM model for SQuAD [40] style extractive QA

wherein start and end tokens of a span is predicted. For this, we need to prepare LayoutLM training data in SQuAD-style format where answer is marked as a span of the text present on the infographic image. We serialize the OCR tokens spotted on each image in the natural reading order. Then we check if a ground truth answer can be found as a subsequence of the serialized text. In cases where an answer has multiple subsequence matches with the serialized text, the first match is taken as the answer span. If no match is found, the particular question is not used finetuning the model. This approach of finding answer spans from answers is inspired by the similar approach used by authors of TriviaQA [24]. The same has been used by authors of DocVQA as well for finetuning a BERT QA model for span prediction. Unlike substring matching in TriviaQA we look for subsequence matches as proposed by DocVQA authors. Substring matches can result in many false matches. Since InfographicVQA has lot of numeric answers false matches are even more likely. For example if the answer is ‘3’ (the most common answer in InfographicVQA dataset), if we go by substring matching, it will match with a 3 in ‘300’ and ‘3m’.

D. Additional Qualitative Examples

In Figure D.1– Figure D.7, we show qualitative examples covering multiple types of answers, evidences and operations which we discuss in Section 3.3 of the main paper. These results supplement the qualitative results we show in Section 5.2 in the main paper.

References

- [1] Amazon Texttract. <https://aws.amazon.com/texttract/>. Accessed: 2021-08-16.
- [2] Huggingface’s Models. <https://huggingface.co/models>. Accessed: 2021-08-16.
- [3] stacked attention networks for image question answering.
- [4] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering, 2017.
- [5] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, 2018.
- [6] Srikanth Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R. Manmatha. Docformer: End-to-end transformer for document understanding, 2021.
- [7] Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluis Gomez, Marcal Rusinol, Ernest Valveny, C.V. Jawahar, and Dimosthenis Karatzas. Scene Text Visual Question Answering. In *ICCV*, 2019.
- [8] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.
- [9] M. A. Borkin, A. A. Vo, Z. Bylinskii, P. Isola, S. Sunkavalli, A. Oliva, and H. Pfister. What Makes a Visualization Memorable? *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2306–2315, 2013.
- [10] Z. Bylinskii, Sami Alsiekh, Spandan Madan, A. Recasens, Kimberli Zhong, H. Pfister, F. Durand, and A. Oliva. Understanding infographics through textual and visual tag prediction. *ArXiv*, abs/1709.09215, 2017.
- [11] R. Chaudhry, S. Shekhar, U. Gupta, P. Maneriker, P. Bansal, and A. Joshi. Leaf-qa: Locate, encode attend for figure question answering. In *WACV*, 2020.
- [12] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. UNITER: learning universal image-text representations. *CoRR*, abs/1909.11740, 2019.
- [13] Michael A. A. Cox and Trevor F. Cox. *Multidimensional Scaling*, pages 315–347. Springer Berlin Heidelberg, 2008.
- [14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *ACL*, 2019.
- [15] Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. DROP: A Reading Comprehension Benchmark Requiring Discrete Reasoning Over Paragraphs. In *NAACL-HLT*, 2019.
- [16] Łukasz Garnarek, Rafał Powalski, Tomasz Stanisławek, Bartosz Topolski, Piotr Halama, and Filip Graliński. Lambert: Layout-aware language modeling using bert for information extraction. *arXiv preprint arXiv:2002.08087*, 2020.
- [17] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh. Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering. In *CVPR*, 2017.
- [18] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *CVPR*, 2018.
- [19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *CVPR*, 2017.
- [20] Himanshu. Detectron2 for document layout analysis. <https://github.com/hpanwar08/detectron2.git>, 2020.
- [21] Ronghang Hu, Amanpreet Singh, Trevor Darrell, and Marcus Rohrbach. Iterative answer prediction with pointer-augmented multimodal transformers for textvqa. In *CVPR*, 2020.
- [22] Drew A. Hudson and Christopher D. Manning. GQA: a new dataset for compositional question answering over real-world images. *CoRR*, abs/1902.09506, 2019.
- [23] Tanuj Jain, Christopher Lennan, Zubin John, and Dat Tran. Imagededup. <https://github.com/idealo/imagededup>, 2019.
- [24] Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *ACL*, 2017.
- [25] Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. DVQA: Understanding Data Visualizations via Question Answering. In *CVPR*, 2018.

- [26] Samira Ebrahimi Kahou, Vincent Michalski, Adam Atkinson, Ákos Kádár, Adam Trischler, and Yoshua Bengio. Figureqa: An annotated figure dataset for visual reasoning. *arXiv preprint arXiv:1710.07300*, 2017.
- [27] Vahid Kazemi and Ali Elkursh. Show, ask, attend, and answer: A strong baseline for visual question answering. *arXiv preprint arXiv:1704.03162*, 2017.
- [28] Aniruddha Kembhavi, Minjoon Seo, Dustin Schwenk, Jonghyun Choi, Ali Farhadi, and Hannaneh Hajishirzi. Are You Smarter Than A Sixth Grader? Textbook Question Answering for Multimodal Machine Comprehension. In *CVPR*, 2017.
- [29] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *Int. J. Comput. Vision*, 2017.
- [30] Nathan Landman. Towards abstractive captioning of infographics. Master’s thesis, Massachusetts Institute of Technology, Massachusetts Institute of Technology, 2018.
- [31] Liunian Harold Li, Mark Yatskar, Da Yin, C. Hsieh, and Kai-Wei Chang. VisualBERT: A Simple and Performant Baseline for Vision and Language. *ArXiv*, abs/1908.03557, 2019.
- [32] Yulin Li, Yuxi Qian, Yuchen Yu, Xiameng Qin, Chengquan Zhang, Yan Liu, Kun Yao, Junyu Han, Jingtuo Liu, and Errui Ding. Structext: Structured text understanding with multimodal transformers, 2021.
- [33] Min Lu, Chufeng Wang, Joel Lanir, Nanxuan Zhao, Hanspeter Pfister, Daniel Cohen-Or, and Hui Huang. Exploring Visual Information Flows in Infographics. In *ACM CHI*, 2020.
- [34] Ben Mabey. pyLDAvis. <https://github.com/bmabey/pyLDAvis>, 2021.
- [35] Spandan Madan, Zoya Bylinskii, Matthew Tancik, Adrià Recasens, Kimberli Zhong, Sami Alsheikh, Hanspeter Pfister, Aude Oliva, and Fredo Durand. Synthetically trained icon proposals for parsing and summarizing infographics. *arXiv preprint arXiv:1807.10441*, 2018.
- [36] Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. DocVQA: A Dataset for VQA on Document Images. In *WACV*, 2020.
- [37] Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. OCR-VQA: Visual question answering by reading text in images. In *ICDAR*, 2019.
- [38] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*. 2019.
- [39] Rafal Powalski, Łukasz Borchmann, Dawid Jurkiewicz, Tomasz Dwojak, Michał Pietruszka, and Gabriela Palka. Going full-tilt boogie on document understanding with text-image-layout transformer. *arXiv preprint arXiv:2102.09550*, 2021.
- [40] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, 2016.
- [41] Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *LREC 2010 Workshop on New Challenges for NLP Frameworks*, 2010.
- [42] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *NeurIPS*. 2015.
- [43] Amanpreet Singh, Vedanuj Goswami, Vivek Natarajan, Yu Jiang, Xinlei Chen, Meet Shah, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. MMF: A multimodal framework for vision and language research. <https://github.com/facebookresearch/mmf>, 2020.
- [44] Amanpreet Singh, Vedanuj Goswami, and D. Parikh. Are we pretraining it right? Digging deeper into visio-linguistic pretraining. *ArXiv*, abs/2004.08744, 2020.
- [45] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Devi Parikh, and Marcus Rohrbach. Towards VQA Models That Can Read. In *CVPR*, 2019.
- [46] Weijie Su, X. Zhu, Y. Cao, B. Li, Lewei Lu, Furu Wei, and Jifeng Dai. ViLBert: Pre-training of generic visual-linguistic representations. *ArXiv*, abs/1908.08530, 2020.
- [47] Ryota Tanaka, Kyosuke Nishida, and Sen Yoshida. Visualmrc: Machine reading comprehension on document images. *arXiv preprint arXiv:2101.11272*, 2021.
- [48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *NeurIPS*. 2017.
- [49] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface’s transformers: State-of-the-art natural language processing, 2020.
- [50] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. <https://github.com/facebookresearch/detectron2>, 2019.
- [51] Y. Wu, M. Schuster, Z. Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, M. Krikun, Yuan Cao, Q. Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, Taku Kudo, H. Kazawa, K. Stevens, G. Kurian, Nishant Patil, W. Wang, C. Young, J. Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, G. S. Corrado, Macduff Hughes, and J. Dean. Google’s neural machine translation system: Bridging the gap between human and machine translation. *ArXiv*, abs/1609.08144, 2016.
- [52] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. LayoutLM: Pre-training of Text and Layout for Document Image Understanding. *ACM SIGKDD*, Jul 2020.
- [53] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang,

- Wanxiang Che, et al. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. *arXiv preprint arXiv:2012.14740*, 2020.
- [54] Semih Yagcioglu, Aykut Erdem, Erkut Erdem, and Nazli Ikizler-Cinbis. RecipeQA: A Challenge Dataset for Multimodal Comprehension of Cooking Recipes. In *EMNLP*, 2018.
- [55] Zhengyuan Yang, Yijuan Lu, Jianfeng Wang, Xi Yin, Dinei Florencio, Lijuan Wang, Cha Zhang, Lei Zhang, and Jiebo Luo. TAP: Text-Aware Pre-training for Text-VQA and Text-Caption, 2020.
- [56] Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. PubLayNet: largest dataset ever for document layout analysis. In *ICDAR*, 2019.

Stage 1 instructions	Stage 2 instructions
<p>1. You need to add questions and corresponding answers based on the given image.</p> <p>2. Make sure the questions you ask can be answered purely based on the content in the image</p> <p>3. Try to minimize questions based on textual content. Frame questions which require you to connect multiple elements such as text, structured elements like tables, data visualizations and any other visual elements.</p> <p>4. On infographics with numerical data, try to frame questions which require one to do basic arithmetic, counting or comparisons</p> <p>5. You are allowed to reject the image if,</p> <ul style="list-style-type: none"> (a) bad resolution images/ illegible text (b) image is an infographic template with dummy text (c) content is almost entirely in a non English language <p>6. The text which forms your answer must be</p> <ul style="list-style-type: none"> (a) found verbatim in the image as a contiguous sequence of tokens in the reading order (b) found verbatim in the question as a contiguous sequence of tokens (c) formed by multiple text pieces where each ‘piece’ is found verbatim in the image as contiguous sequence of text tokens. In such a case when you add the answer separate each item by a comma and a white space. (d) a number such as 2, 2.2, 2/2 etc.. 	<p>1. You need to enter answer for the questions shown based on the given image</p> <p>2. if you cannot find answer to the question based on the image, flag the question as “cant answer”</p> <p>3. For each question add the following question-answer types appropriately. There are three categories of question-answer types - Evidence, Operation, and Answer-source. It is possible that a question can have more than one source of answer, more than one type of operation, or more than one type of evidence associated with it.</p> <p>(a) Answer source : Following are different types of sources possible:</p> <ul style="list-style-type: none"> i. Image-Span: answer is found verbatim in the image as a contiguous sequence of tokens in the reading order ii. Question-Span: similar to Image-span but found from question. iii. Multi-Span: formed by multiple text pieces where each ‘piece’ is found verbatim in the image as contiguous sequence of text tokens (i.e, each piece is an Image-span). iv. Non-extractive: answer is a numerical answer and is not found verbatim on the text on the image or the question. <p>(b) Evidence : Following are different types of evidences possible:</p> <ul style="list-style-type: none"> i. Text: answer is derived purely by reading text found in the image. ii. Table/List: finding answer requires one to understand a tabular or list type structure iii. Visual/Layout: requires one to look for visual aspects(colors, names of objects etc.) or layout of the image to arrive at the answer. iv. Figure: requires understanding a figure, a plot, a visualization or a schematic. v. Map: answer is based on a geographical map <p>(c) Operation : if answering the question requires one of the following discrete operations:</p> <ul style="list-style-type: none"> i. Counting: requires to count something ii. Arithmetic: requires to do basic arithmetic operations (sumsubtractmultiplydivide) to find the answer. iii. Sorting: requires to find sort numbers or need to compare numbers

Table A.1: **Annotation instructions.** In addition to the written instructions, workers were trained on the process through webinars. During the webinars, each instruction and annotation policy was explained with the help of multiple examples. This helped the workers get familiarize with the kind of questions and answers that are allowed.

List of documents to annotate

Document	Action
23.covid19-sci.png (checked out)	✓
4.clothing-masks-infographic--web---part-1.png (checked out)	✓
104.covid-19-flyer-what-you-need-to-know-resized.png	✗
21.testing-full20size-pcr-antibody.png	✗
58.avoiding-covid-19-update-e44657.png	✗
55.who-workplace-health-110_slide-2-1200px.png	✗
78.idea.int2040int_idea2028229.png	✗
7.coronavirus-and-hiv-infographic-by-avert.png	✗
39.covid19-infographic-image.png	✗
52.covid19-symptoms.png	✗
85.covid-19-pui-milk-in-neonatal-settings2-309x400.png	✗
1.clothing-masks-infographic---web---part-2.png	✗
13.covid_infographic_4-6-20.png	✗
97.impact-retail-sector.png	✗
25.620nxnjg200003fa.png.pagespeed.ic.rjpxpjzqdq.png	✗

Figure A.1: **Images queue for question-answer collection.** The screenshot shows interface from which a worker picks an image for question-answer collection (stage 1 of annotation). It shows the list of all images in the system that are yet to be annotated, and the images that are already opened (checked out) by the particular user. In this case there are two images that are being checked out; the two images shown at the top of the list. Workers were allowed to check out at most 5 images. This feature allows workers to keep documents in pending if they are in doubt.

The interface consists of two main panes. The left pane displays a news article titled "COVID-19 DISORDER TRACKER: SPOTLIGHT" with the date "5-11 APR 2020". The article's title is "Healthcare Under Workers Siege". It includes a map of the world showing incidents of violence against healthcare workers, a bar chart titled "From Protesters to Targets" showing the number of events over time, and a donut chart showing the distribution of violence types. The right pane is titled "Questions" and contains a list of questions with input fields and buttons for "Cancel" and "Add".

Question	Answer
In the bar chart what color is used to represent peaceful protest - green, blue, orange or red?	orange
No answers yet.	
Approximately how many events involving healthcare workers occurred in the last week of April?	40
68% of violence targeting healthcare workers reported in which country?	India
According to the doughnut chart which country reported second highest number of political violence incidents targeting healthcare workers?	Philippines
How many countries are shown in the doughnut chart given below?	8
Which is the southern most country in Africa?	

Figure A.2: **First stage of annotation.** Questions and answers are collected at this stage. Image is shown on the left pane with options to zoom, rotate etc. Questions and answers are added on the right pane.



Figure A.3: **Second stage of annotation.** Additional answers and question-answer types are collected for validation and test splits. For one of the questions, answer cannot be found from the image alone and hence the worker assigned a “cant answer” flag. Such questions are removed from the dataset.

No.	Topic
1	cost lead increase system non risk energy reduce cause clean
2	war violence symptom domestic potential die injury mil acquire birth
3	health person white black police department doctor respiratory smith officer
4	child food water parent potential eat drink essential green sugar already
5	death woman age man old adult love likely statistic rate
6	country high account say month report change global survey event
7	social medium job value program find direct authority salary candidate
8	first purchase call sport still house kid name bring early
9	case university point physical idea language mass brain thought presentation
10	fire act min sunday encounter concentration daily active th monthly
11	paper common check photo add type virus print christmas present
12	game mobile internet app olympic london medal online device mm_mm
13	public right patient human goal influence earth plant face individual
14	help free american likely provide need support contact tip hand
15	company school design content employee college technology create offer audience
16	new state top city rank york art west east california
17	business customer service population sale product small software increase investment
18	force industry car line waste register decrease driver victim throw
19	year world people day make time com average number source
20	user use facebook share site video post google search worldwide

Table B.1: **Top 20 topics in InfographicVQA found using LDA.** We used text tokens spotted on the images for topic modelling.

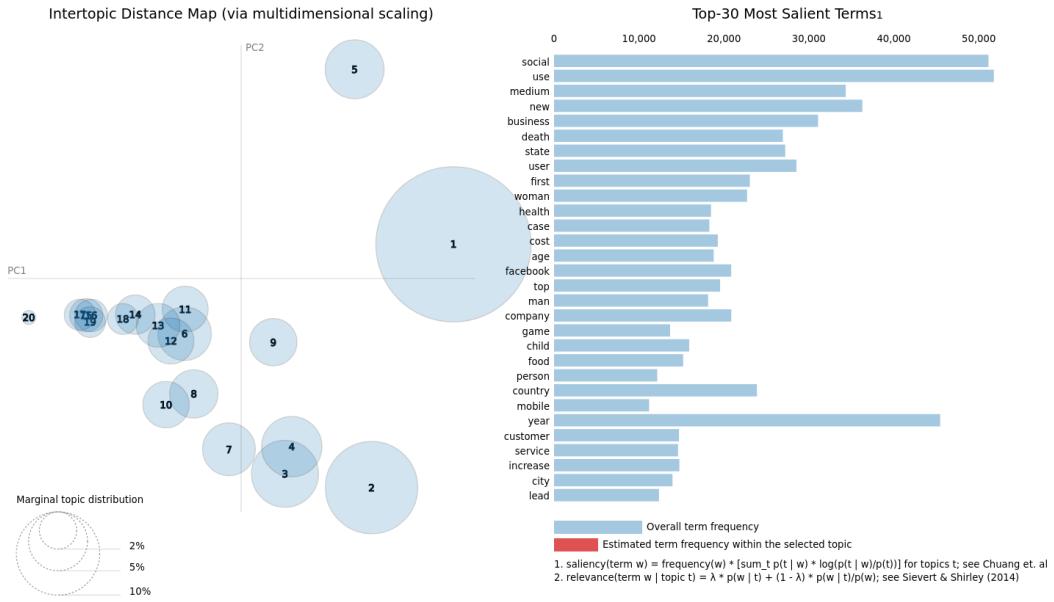


Figure B.1: **Visualization of the top 20 topics in InfographicVQA dataset.** We used LDA to find the topics. On the left is an inter topic distance map where each circle represent a topic. The area of the circles is proportional to the overall relevance of the topic. Distance between two topics are computed using Jensen–Shannon divergence. The distances are then mapped to two dimensional space using Multidimensional Scaling [13]. On the right we show top 30 most salient terms(most prevalent terms in the entire corpus) among the text present on the images. This diagram was created using pyLDAvis tool.

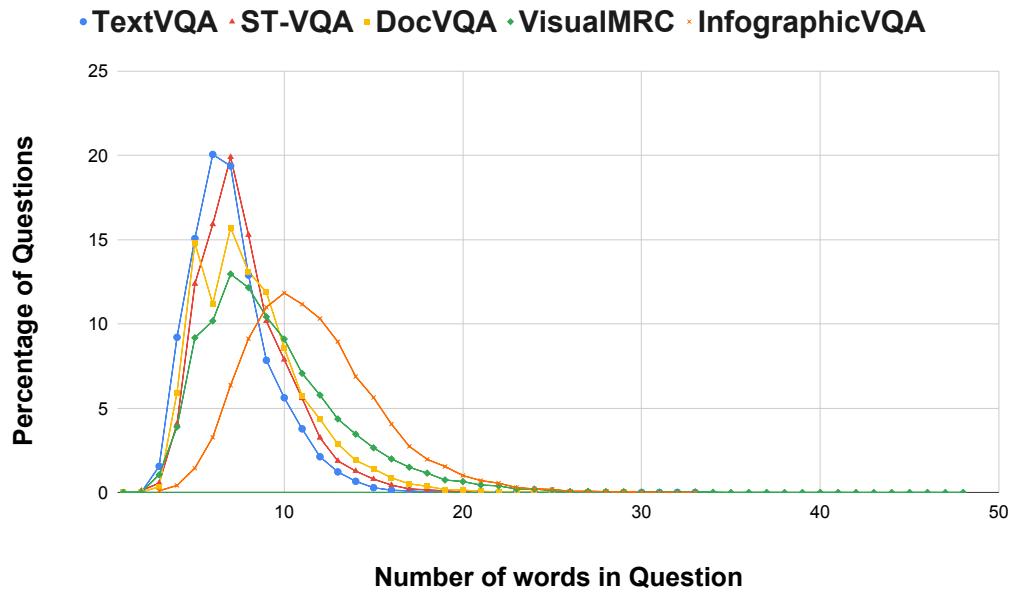


Figure B.2: **Percentage of questions with a particular length.** Compared to other similar datasets, questions in InfographicVQA are longer on average. Average question length is 11.54 (Table 2 in the main paper), which is highest among similar datasets including VisualMRC, which is an abstractive QA dataset.

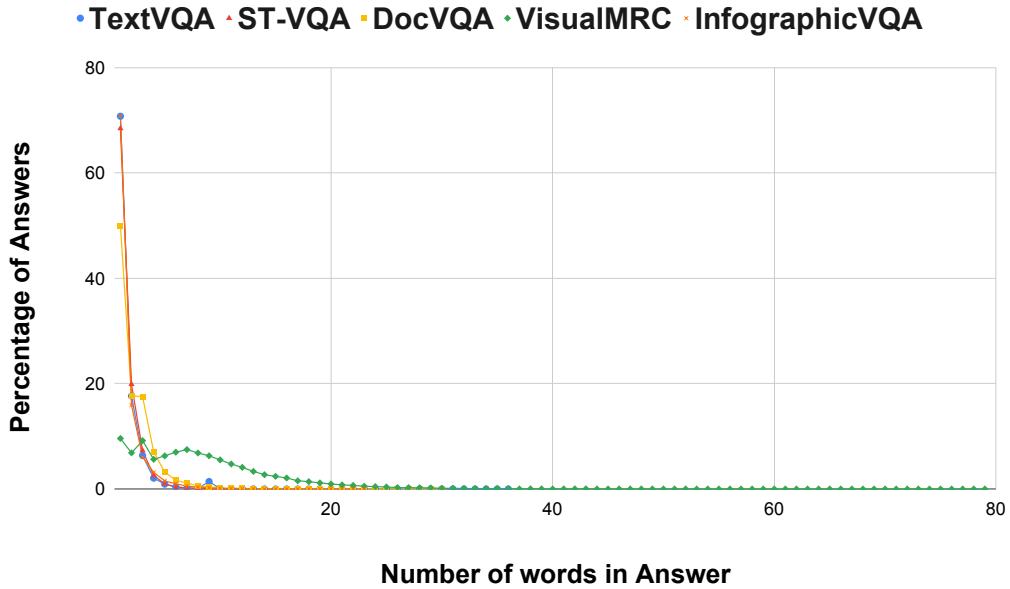


Figure B.3: **Percentage of answers with a particular length.** Answers in InfographicVQA are shorter compared to most of the similar datasets. More than 70% of the answers have only single word in it and more than 85% have at most 2 words in it. This is expected since the questions are asked on data presented on infogrpahics, which is mostly numerical data.

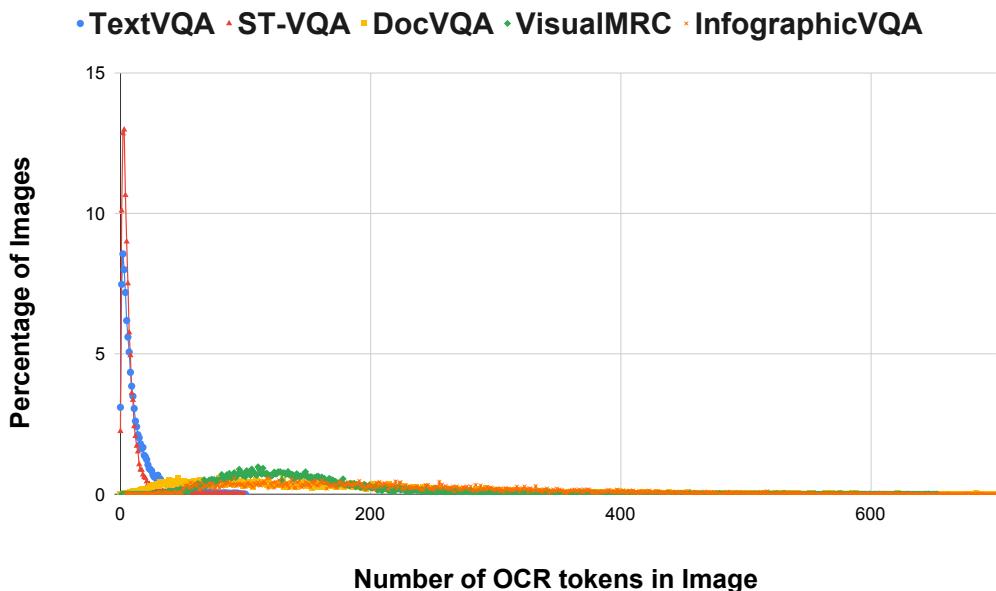


Figure B.4: **Percentage of images with a particular number of text tokens on it.** Average number of text tokens per image is highest in InfographicVQA (Table 2 in the main paper). It can be seen from the plot that InfographicVQA has a flatter curve with a longer tail.

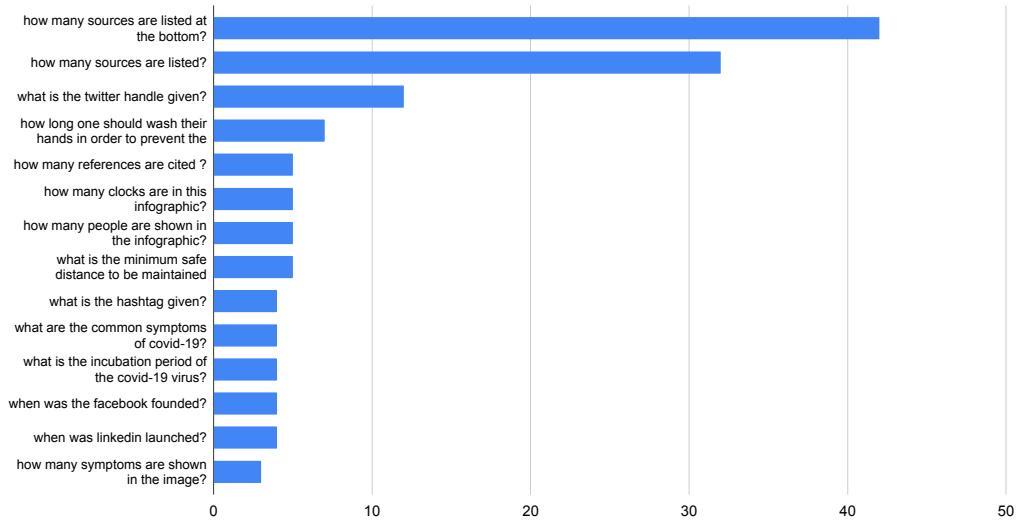


Figure B.5: **Top 15 questions.** A majority of commonly occurring questions concern with counting.

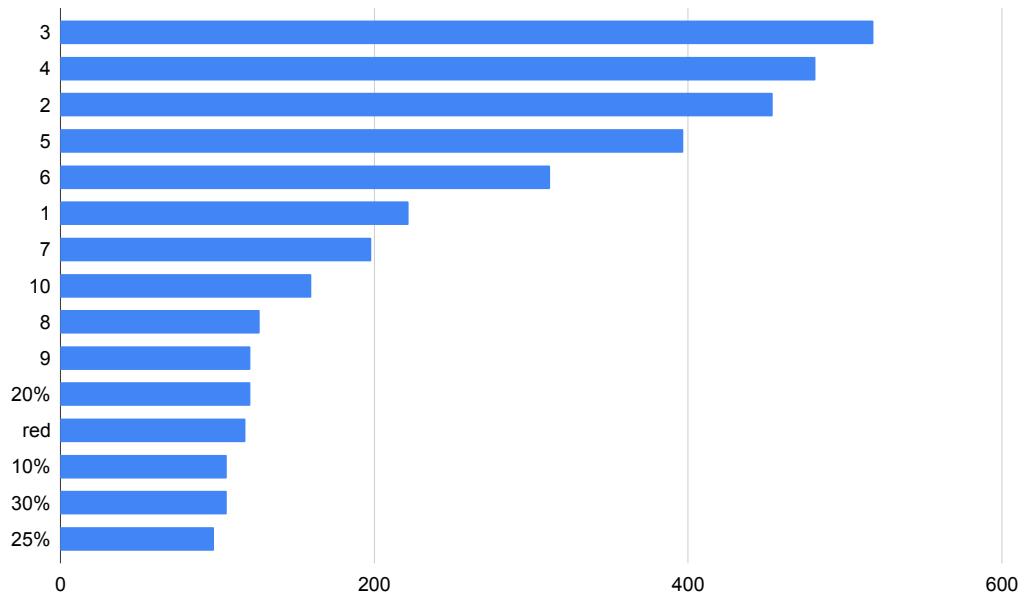


Figure B.6: **Top 15 answers.** Almost all of the top answers are numeric answers.

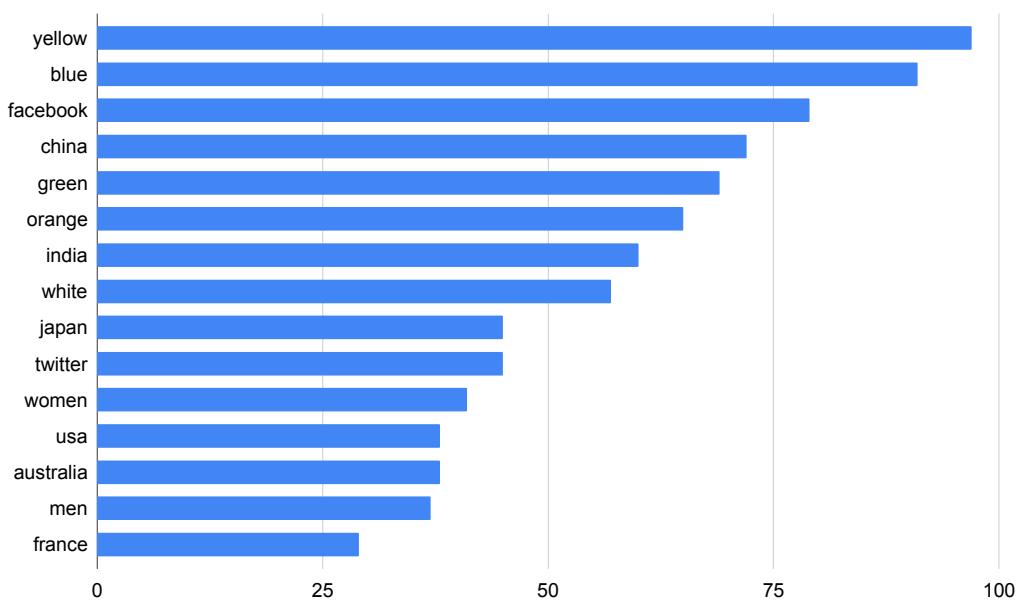
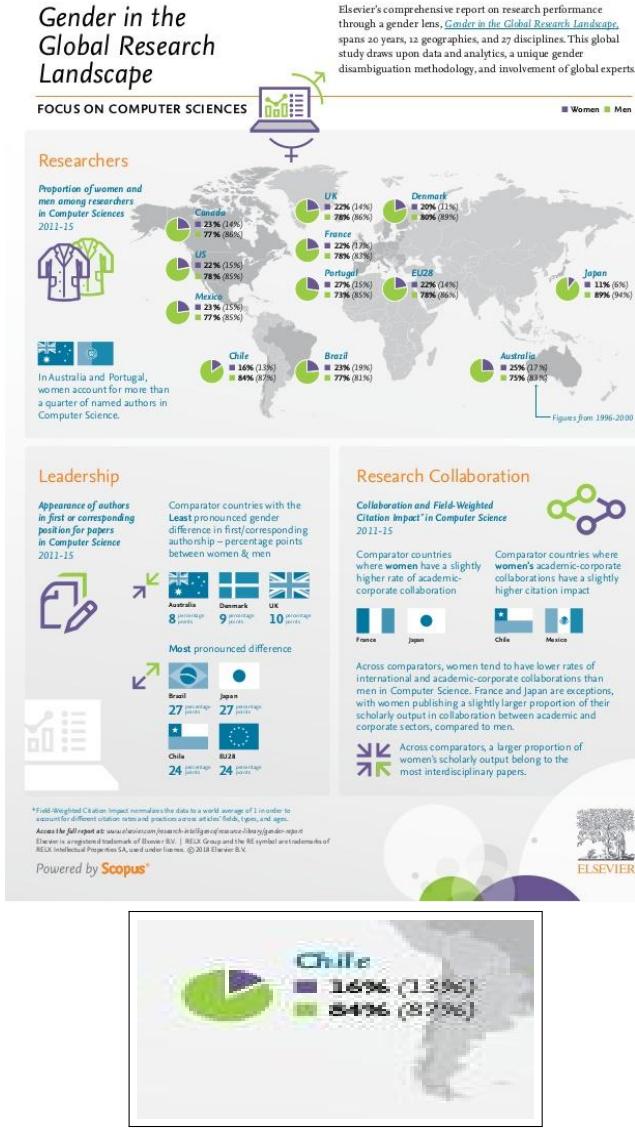


Figure B.7: **Top 15 non-numeric answers.** Non numeric answers are dominated by names of countries, names of colors and names of social media platforms.



Q: what percent of researchers in Chile were men in the duration of 2011-15?

GT: [84%, 84]

LayoutLM: 23%

M4C: 23%

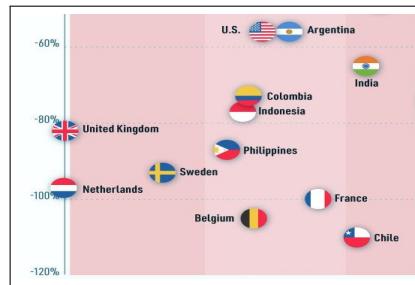
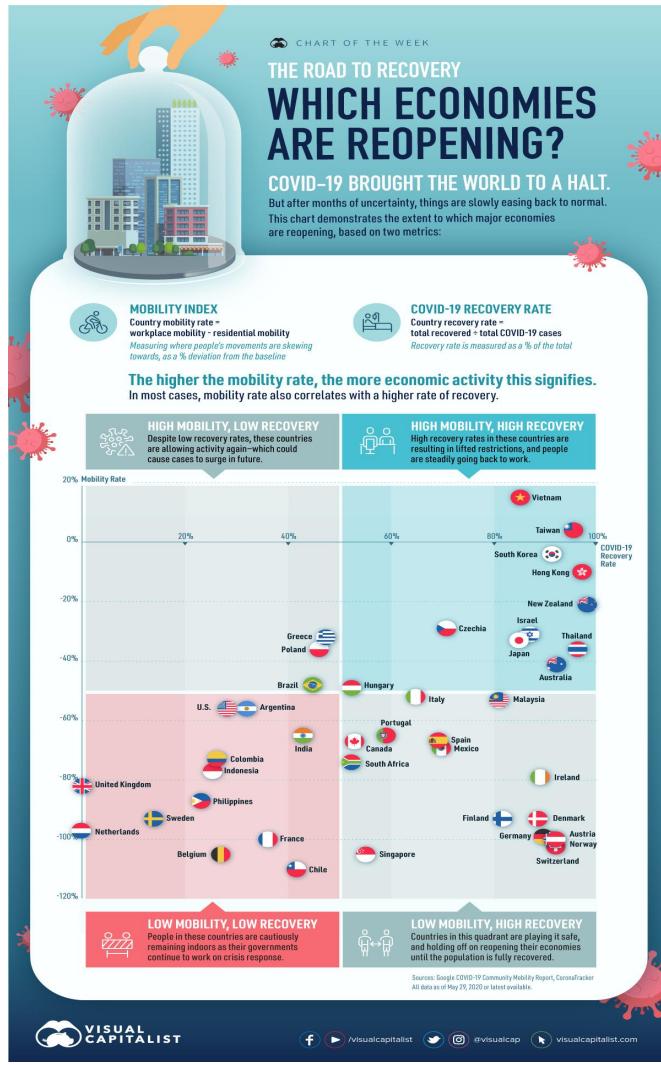
Human: 84%

Answer-source: Image-span

Evidence: Map

Operation: none

Figure D.1: **Using color codes and information on a Map to arrive at answer.** To answer this question, models require to understand that the blue color correspond to women and then pick the number corresponding to the blue color from the data given for Chile. Both the baseline models get this question wrong. Note that here there are two valid answers (GT), one added during the first stage of annotation and the other during the second stage.



Q: Which quadrant does the country India fall into, blue, pink, or gray?

GT: pink

LayoutLM: country

M4C: pink

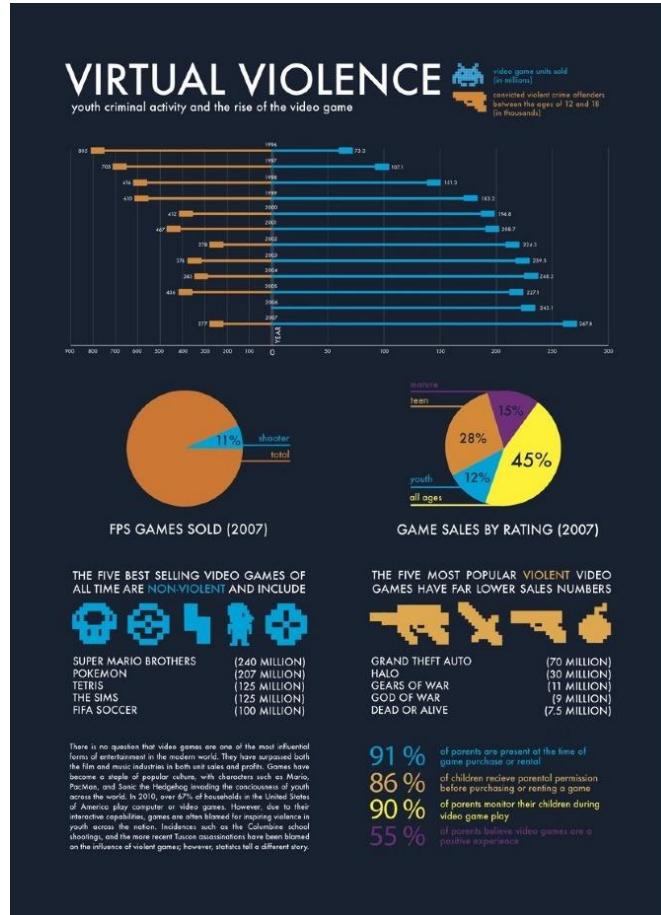
Human: pink

Answer-source: Question-span

Evidence: Figure Visual/Layout

Operation: none

Figure D.2: Answer is a color name which is given as a multiple choice option in the question. To answer this question, a model needs to first locate "India" on the image and then identify the background color there. M4C gets this question correct.



Q: Which are the top 2 best selling non violent video games of all time?

GT: super mario brothers, poke-
mon
LayoutLM: super mario brothers
M4C: instagram youth
Human: super mario brothers,
pokemon

Answer-source: Multi-span
Evidence: Table/List
Operation: Sorting

Figure D.3: **Multi-Span answer.** Multi-span answer type allows us to include questions where answer is formed by multiple single ‘span’s. In this example top 2 items in a category need to be found. This can only be answered if we allow answers containing multiple spans. Since the LayoutLM-based model we train for extractive QA can handle only single spans, it gets first part of the answer correct.



Q: How many championships has Kobe Bryant won?

GT: 5

LayoutLM: 5

M4C: 5

Human: 5

Answer-source: Non-extractive

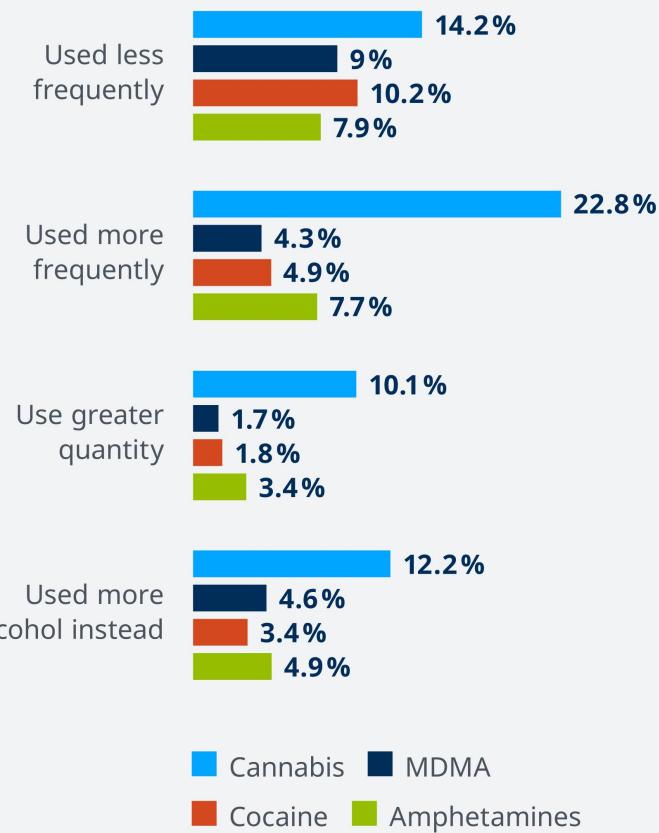
Evidence: Figure

Operation: Counting

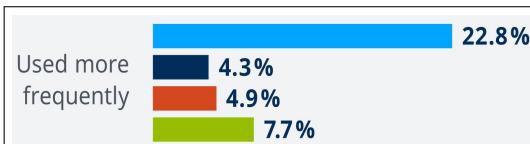
Figure D.4: Counting symbols/markers to find an answer. Both the models get the answer correct for this question that require one to count the yellow squares next to "CHAMPIONSHIPS".

How the first COVID-19 lockdowns have affected European drug use

European Web Survey on Drugs, April-May 2020



Source: EWS / EMCDDA



Q: Which drug was used more frequently during lockdown, MDMA, Cocaine, Cannabis, or Amphetamines?

GT: cannabis

LayoutLM: cannabis

M4C: cocaine

Human: cannabis

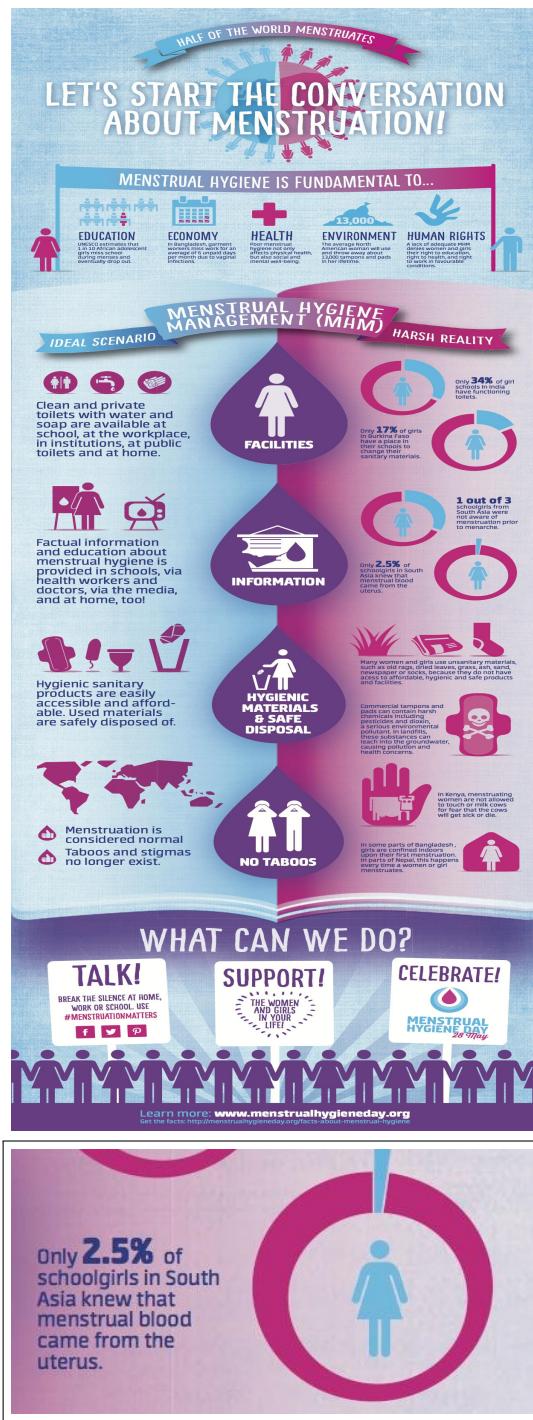
Answer-source: Question-span

Image-span

Evidence: Figure

Operation: Sorting

Figure D.5: **Sorting values shown in a bar chart.** In this question, answer is a span of question (Question-span) and a span of the text on the image (Image-span) as well. The largest among the given items is explicit in the bar chart representation. Alternatively the same can be found by finding the largest by comparing the numbers. Hence 'Sorting' is added as the Operation.



Q: What % of schoolgirls in South Asia do not know that menstrual blood comes from the uterus?

GT: [97.5, 97.5%]

LayoutLM: 2.5%

M4C: 25

Human: 97.5%

Answer-source: Non-extractive

Evidence: Text

Operation: Arithmetic

Figure D.6: **Question requiring arithmetic operation.** To answer this question, the given percentage value needs to be subtracted from 100. Both the models fail to get the answer correct.



Q: Playing against which country did he reach the most number of his milestone runs?

GT: sri lanka

LayoutLM: bangladesh

M4C: pakistan

Human: sri lanka

Answer-source: Image-span

Evidence: Text Figure

Operation: Counting Sorting

Figure D.7: **Performing multiple discrete operations.** Here the context required to find the answer spans the entire image. Hence we do not show a crop of the image in the inset. This question requires a model to do Counting — count number of milestone runs scored against each country and then perform Sorting — find the country against which the player scored most milestone runs.