Chart-to-Text: A Large-Scale Benchmark for Chart Summarization

Shankar Kantharaj[♣,*], **Rixie Tiffany Ko Leong**[♦,*], **Xiang Lin**[♦,*], **Ahmed Masry**[♣,*]
Megh Thakkar[♦,*], **Enamul Hoque**[♣], **Shafiq Joty**[♦♠]
[♣]York University, Canada, [♦]Nanyang Technological University, Singapore
[♠]Salesforce Research Asia, Singapore
{shankark, masry20, enamulh}@yorku.ca
{rleong007, linx0057, srjoty}@ntu.edu.sg
megh.1211@gmail.com

Abstract

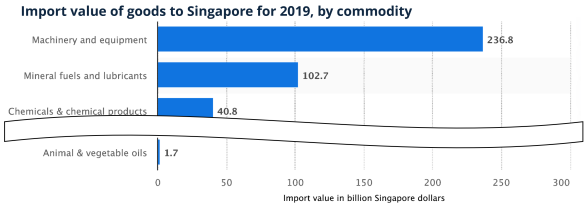
Charts are commonly used for exploring data and communicating insights. Generating natural language summaries from charts can be very helpful for people in inferring key insights that would otherwise require a lot of cognitive and perceptual efforts. We present Chart-to-text, a large-scale benchmark with two datasets and a total of 44,096 charts covering a wide range of topics and chart types. We explain the dataset construction process and analyze the datasets. We also introduce a number of state-of-the-art neural models as baselines that utilize image captioning and data-to-text generation techniques to tackle two problem variations: one assumes the underlying data table of the chart is available while the other needs to extract data from chart images. Our analysis with automatic and human evaluation shows that while our best models usually generate fluent summaries and yield reasonable BLEU scores, they also suffer from hallucinations and factual errors as well as difficulties in correctly explaining complex patterns and trends in charts.

1 Introduction

Data visualizations such as bar charts, line charts, and pie charts are very popular for presenting quantitative data. Often people use such charts to get important insights from data and make informed decisions. However, it is well-known that inferring key insights from the charts can be quite challenging and time-consuming, as it may require a lot of cognitive and perceptual efforts (Pérez-Echeverría et al., 2018; Whitaker and Jacobbe, 2017).

Automatic chart summarization is a task where the goal is to explain a chart and summarize key takeaways from it in natural language. Chart summarization has several key benefits and potential applications. First, chart summaries can help people identify key insights from charts that they might have missed otherwise. In a study on a chart corpus, Carberry et al. (2006) found that chart authors often failed to convey key insights from charts in their corresponding textual captions. Thus, automatic summarization could help authors write effective reports and articles on data facts by suggesting explanatory texts. Similarly, readers could benefit from such summaries, as studies have found that captions help readers find important points by explaining visually prominent features in charts (Kim et al., 2021). Chart summarization offers another important benefit of making charts more accessible to people who are visually impaired since they can use screen readers to understand what is being presented in the chart (Ferres et al., 2013). Finally, the generated summaries can be leveraged for indexing documents containing charts to improve information retrieval algorithms (Li et al., 2013).

Despite its numerous benefits and applications, the chart summarization problem has not received much attention in the NLP community. Early approaches relied on template-based text generation methods that combine statistical techniques and planning-based architecture (Reiter, 2007) to generate captions from bar and line charts (Fasciano and Lapalme, 1996; Mittal et al., 1998; Green et al.,



Gold: In 2019, Singapore imported approximately 236.8 billion Singapore dollars worth of machinery and equipment, making it the country's largest import commodity by value. This was followed by the import of mineral fuels and lubricants, valued at 102.7 billion Singapore dollars.
TAB-T5: Machinery and equipment was the most valuable commodity for Singapore in 2019, with an import value of 236.8 billion Singapore dollars. Mineral fuels and lubricants were the second most valuable commodity for Singapore, with an import value of 102.7 billion Singapore dollars.

Figure 1: An example chart-summary pair from our Benchmark and the output from one of the best models (TAB-T5).

*Equal contribution. Listing order is based on the alphabetical ordering of author surnames.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 4005 - 4023

2004; Demir et al., 2012). Recently, researchers considered data-driven neural models for describing tabular data (Mei et al., 2016; Gong et al., 2019). However, compared to tables, charts serve a different communication goal, and so is the chart-to-text problem. Unlike tables which simply list raw data, charts create visual representation of data that can draw a reader's attention to various prominent features such as trends and outliers (Kim et al., 2021). For example, a line chart may depict an important trend whereas a scatterplot may visually communicate correlations and outliers. Existing table-to-text approaches are not designed to explain such visually salient chart features in summaries.

There are two main impediments to addressing the chart summarization task. First, the lack of large-scale datasets makes it difficult to solve the task using data-driven neural models. Second, there are no strong baselines that utilize the latest advances in neural text generation tasks. Obeid and Hoque (2020) made an initial attempt to address this problem with a dataset and a model that utilizes a Transformer (Vaswani et al., 2017) architecture. However, their dataset was built by collecting a small set of charts (8,305) from a single source covering only two types of charts (bar and line). Also, their approach does not exploit the recent advances in large-scale language model pretraining, which has been shown to be very beneficial for many vision and language tasks (Devlin et al., 2019; Touvron et al., 2021). To our knowledge, there is no large-scale benchmark with a wider range of topics from multiple sources, covering many different chart types, and with models that employ large-scale pretraining.

In this work, we present a large-scale benchmark for chart-to-text with two datasets consisting of 44,096 charts covering a broad range of topics and a variety of chart types. We introduce two variations of the problem. The first variation assumes that the underlying data table of a chart is available, while the other introduces a more challenging and realistic scenario by assuming that the chart is in image format and the underlying table is not available. These two problem scenarios motivated us to adapt a variety of state-of-the-art models that combine computer vision and natural language generation techniques as strong baselines; see Fig. 1 for a sample model output.

Our primary contributions are: (*i*) a new large-scale benchmark covering a wide range of topics and chart types; (*ii*) a set of state-of-the-art neural models which can act as a starting point for other researchers to expand and improve upon; and (*iii*) a series of automatic and human evaluations as well as in-depth qualitative analysis to identify further challenges. Our code and benchmark datasets are publicly available at https://github.com/vis-nlp/Chart-to-text.

2 Related Work

Chart Summarization Early work (Mittal et al., 1998; Ferres et al., 2013) followed a planning-based architecture (Reiter, 2007) and used templates to generate texts. These systems only describe how to read the chart rather than explain key insights conveyed by the chart. Recently, commercial systems such as Quill and Wordsmith[1] as well as research prototypes, *e.g.,* (Cui et al., 2019) and (Srinivasan et al., 2018) computed statistics (*e.g.,* extrema, outliers) to present facts from a dataset. Demir et al. (2012) also compute statistics to generates bar chart summaries in a bottom–up manner to simultaneously construct the discourse and sentence structures. Recently, Chen et al. (2019) used the ResNet (He et al., 2016) to encode the chart image and an LSTM decoder to create the caption.

A key limitation of the above bodies of work is that sentences are generated using predefined templates, which may lack generality and offer little variation in terms of reported insights, grammatical styles and lexical choices compared to data-driven models. Moving beyond template-based summaries, Obeid and Hoque (2020) adapted a transformer-based model on a dataset of 8,305 charts, while Spreafico and Carenini (2020) applied an LSTM based encoder-decoder model on a dataset of 306 chart summaries. Both studies used much smaller datasets and did not consider the computer vision aspects of the problem. Hsu et al. (2021) recently use a CNN+LSTM based image captioning model for scientific figure captioning. In contrast, we focus on the generic chart-to-text problem and train several neural models that combine computer vision and data2text generation.

Data2text Generation Data2text models generate a descriptive summary for a table of records. They have been used for various domain-specific tasks such as summarizing sports data (Barzilay and Lapata, 2005; Wiseman et al., 2017), weather-

[1]Narrative Science Quill; Automated Insights Wordsmith

forecast data (Reiter et al., 2005), recipe generation (Yang et al., 2017) and biography generation (Lebret et al., 2016) as well as open-domain tasks (Parikh et al., 2020; Chen et al., 2020a). Recent methods have primarily used an LSTM-based encoder-decoder architecture (Mei et al., 2016; Lebret et al., 2016; Wiseman et al., 2017). Gong et al. (2019) found that transformers (Vaswani et al., 2017) yielded more fluent and coherent outputs compared to their LSTM counterparts. Others focused on controlling the structure of the summary using a planning approach (Su et al., 2021) as well as generating facts by preforming logical inference over the given table (Chen et al., 2020a,b).

Image Captioning There has been swift progress in image captioning largely due to the availability of large-scale datasets (Agrawal et al., 2019; Chen et al., 2015). Zhang et al. (2021) developed an object detection model to summarize objects in images while Sidorov et al. (2020) utilized texts extracted from images using OCR to generate captions. Unlike images with real-world objects and scenes, charts have marks (*e.g.,* bars, lines) that map quantitative data. This makes the chart-to-text problem different from image captioning.

3 Chart-to-text Datasets

After searching through various sources including news sites, textbooks, and websites containing data facts, we found two suitable sources with sufficiently large numbers and varieties of charts with textual descriptions as we describe below.

3.1 Data Collection

• **Statista** Statista (statista.com) is an online platform that regularly publishes charts on a wide range of topics including economics, market and opinion research. We crawled 34,810 publicly accessible webpages in December 2020, yielding a total of 34,811 charts. For each chart, we took a screenshot of the chart image, downloaded the data table, the title, axis labels and the human-written descriptions about the chart. We classified the charts into two groups based on the number of columns in their underlying data tables: Data tables of *simple* charts have only two columns, whereas *complex* charts involve at least three columns (*e.g.,* stacked or group bar charts, line charts with multiple lines).

• **Pew** The Pew Research (pewresearch.org) publishes data-driven articles about social issues, public opinion and demographic trends. The articles are often accompanied by multiple charts along with high-quality descriptions written by professional editors. We scraped 3,999 publicly accessible pages in January 2021, which gave a total of 9,285 charts. Unlike Statista, the Pew reports do not provide the underlying data tables for most of the charts. Among 9,285 charts, only 143 have underlying data tables. For each chart, we downloaded the chart image, the surrounding paragraphs and the alternative text associated with the image (using the `alt` attribute), if it was available. Like a title, the `alt` text often gives a very short chart description. Finally, we classified the charts into *simple* and *complex* manually since underlying data tables were unavailable.

3.2 Data Annotation

Below we describe two main steps of the data annotation process for each chart: (*i*) identify the relevant summary, and (*ii*) extract data. Additional details of these steps are provided in Appendix A.1.

• **Statista** We chose the first part of the text (from the chart icon to the next heading) as the chart summary. This is based on the observation that the first part provides a succinct summary of the chart while the remaining parts often contain background information (*e.g.,* the history of a company).

Extracting data from the Statista charts was relatively straightforward as the underlying data tables were available. However, most charts (32,660 out of 34,811) did not provide x-axis labels. To assign representative labels for them, we first used regular expressions on the cell values of such a column to see if it represents common entities (*e.g., year*, *location*). Still, there were 7,170 missing labels remaining. We then applied the Wikidata knowledge base (Wik, 2021) to automatically derive an entity type label based on the data values plotted on x-axis. However, sometimes the resulting labels were too generic (*e.g., human*, *business*). Hence, we manually annotated each label by either accepting the entity type label, if it represents the x-axis accurately, or entering a more specific name.

• **Pew** The annotation for Pew was more challenging as often a webpage contains many charts and paragraphs do not explicitly refer to their relevant chart. Also, most charts did not have underlying data tables. To address these challenges, we construct the dataset in three stages (Fig. 2).

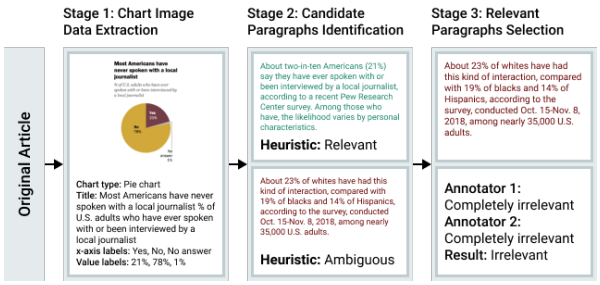


Figure 2: Stages of the Pew dataset construction process.

	Statista		Pew	
Type	Simple	Complex	Simple	Complex
Bar	24,591	5,616	807	5,497
Line	2,646	902	325	2,129
Area	0	0	29	105
Scatter	0	0	0	68
Pie	409	0	325	0
Table	223	424	0	0
Total	27,869	6,942	1,486	7,799

Table 1: Chart type distribution.

(*i*) *Data extraction from chart images:* We first extracted the text from the charts using CRAFT (Baek et al., 2019a,b), a state-of-the-art OCR model. We then extracted the bounding boxes of the detected texts to extract geometric features (*e.g.,* normalized width and height of the text) and used them to train a gradient boosting classifier that categorizes the recognized text into one of the following categories: title, axis labels, legends, and data labels. Since the visual style and structure vary among chart types, we trained a separate classifier for each chart type. We manually labeled 319 examples (171 bar, 68 line, and 80 pie charts) and split them into train, validation, and test splits with 8:1:1 ratios, respectively. Our models achieved a precision of 95.0% overall and 97.6% for title classification on our test set. We then used our models to predict the text roles for the remaining charts in the Pew dataset.

We used the extracted title as the final chart title if there was no associated `alt` text with the chart image. If the `alt` text was available, we took the longer one by comparing it with the extracted title.

(*ii*) *Identification of candidate paragraphs:* We observed that relevant paragraphs tend to appear in close proximity to a given chart and share some content with the chart (*e.g.,* axis labels, data values). We first used this proximity criteria to form a list of candidate paragraphs $\mathcal{L}_c$. Specifically, for each chart, we selected the paragraph adjacent to the chart as well as the five paragraphs before and after it as candidates (maximum of 11 in total).

Next, we used a heuristic-based approach to automatically select a subset of relevant paragraphs $\mathcal{L}_r \subset \mathcal{L}_c$. We estimated the relevance score of each paragraph in $\mathcal{L}_c$ to its corresponding chart as $rel = content \times proximity$, where $content$ takes a weighted sum of the number of tokens matched between the paragraph and the OCR-extracted text (numerical tokens were given a higher weight than lexical tokens as they were better indicators of relevance), and $proximity$ is based on the distance between the chart and the paragraph. If rel exceeds a threshold and some minimum number of lexical and numerical tokens are matched between the paragraph and chart, we consider such a paragraph to be relevant to the chart. We set this threshold empirically and chose it to be aggressively high to prioritize precision over recall. We evaluated the efficacy of our approach against a randomly sampled set of 95 charts and 769 surrounding paragraphs and found a recall of 21.1% and a precision of 100%. Given the perfect precision score, we considered the paragraphs in $\mathcal{L}_r$ to be relevant and to confirm the relevance of the remaining paragraphs, we performed a human study.

(*iii*) *Selection of relevant paragraphs:* We asked crowdworkers on Amazon Mechanical Turk to label how relevant each paragraph is to its chart. A total of 5,478 charts and 13,237 paragraphs were annotated. Each chart received two annotations from two workers. If both workers labeled a paragraph as either completely irrelevant or relevant (partially/completely), we used the label that they agreed upon as the final label.[2] For the remaining 2,888 paragraphs where the workers disagreed, we resolved them through internal annotation.

3.3 Dataset Analysis

Our chart-to-text datasets contain a diverse range of chart types (Table 1). Bar charts make up the majority of the charts both in Statista (87.9%) and Pew (67.9%) for both simple as well as stacked and group bar charts. The next most common type is line charts (10.2% in Statista and 26.4% in Pew).

To analyze the topic distribution, we extracted the topic of each chart using its webpage's metadata (*e.g.,* breadcrumbs, meta-tags). Our datasets cover a broad range of topics including politics, society and health (see Fig. 9 in Appendix A.3).

[2]The overall agreement for the crowd workers was 78.2%.

	Statista		Pew	
Statistic	Simple	Complex	Simple	Complex
#Vocab.	39,191	18,621	9,905	18,067
Avg. Character	295	334	571	635
Avg. Token	54	61	110	124
Avg. Sentence	2.56	2.62	3.84	4.27

Table 2: Chart-to-text dataset statistics.

Content Level	Statista	Pew
Visual encodings	32.03%	0.98%
Statistical and comparative	**50.00%**	**54.63%**
Perceptual and cognitive	8.98%	30.49%
Contextual and domain-specific	10.94%	12.93%

Table 3: Distribution of different types of semantic content.

The topics in Statista are more evenly distributed than the ones in Pew, which is dominated by *U.S. Politics & Policy* (45.4%).

Table 2 presents basic linguistic statistics about the datasets. The summaries in Pew are about twice as long as the those in Statista, in terms of average character, token and sentence count. Unsurprisingly, *complex* charts generally have longer summaries than their *simple* counterparts.

We further analyzed the semantic content of the summaries using 100 randomly sampled chart-summary pairs from each dataset. Table 3 shows the distribution of sentences across the four main types of semantic content.[3] We notice that *statistical and comparative* information (*e.g.,* min, max, avg.) is the most common type of content in both datasets. Summaries in Pew tend to report more insights that require more *perceptual and cognitive* efforts (*e.g.,* trends and causal relations) which are arguably more challenging to generate compared to simple statistics. Both datasets contain comparable proportions of sentences covering *contextual and domain-specific* information. Unlike Statista, Pew summaries rarely explain the chart types and encodings (*e.g.,* what do the `x-` and `y-` axes represent).

We randomly selected 70%, 15%, and 15% of the datasets to create the corresponding train, test and validation splits, respectively.

4 Chart-to-text Baseline Models

Problem Definition We consider two variations of the chart-to-text problem. In the first variation, we assume that the underlying data table of the chart is available, where the dataset can be represented as a set of 4-element tuples $\mathcal{D} = \{\langle C, T, M, S\rangle_n\}_{n=1}^{|\mathcal{D}|}$ with C, T, M and S representing the chart image, data table, metadata and textual summary, respectively. For each cell in the data table T, we have the following information: (*i*) the string value, (*ii*) the row and column positions, and (*iii*) whether it is a header cell or not. The metadata $M = (C_{\text{title}}, C_{\text{type}}, C_{\text{labels}})$ consists of the title, type (*e.g.,* bar, line) and axis labels.

In the second variation, we assume that the data table is not available which makes the problem more challenging as well as realistic because most charts online are in image format and do not have the underlying data tables. For a given input $X = \langle C, T, M\rangle$ or $\langle C, M\rangle$, our goal is to generate a textual description $\hat{S}$ which is a good summary of the chart according to a set of evaluation measures.

We consider three categories of models to tackle the task. The first category is image captioning models, where the task is formulated as generating a textual description for the given chart image. The second category is data-to-text models, which rely on the underlying data tables of the charts to produce the corresponding descriptions. Finally, we consider a combination of vision and text models, where the models first extract the text using the CRAFT OCR model (Baek et al., 2019b) and then train with a data-to-text setup. We present three categories of models below (hyperparameter settings for all the models are provided in Appendix A.3).

4.1 Image Captioning Models

We develop over the Show, Attend, and Tell (SAT) model (Xu et al., 2015) to probe the effectiveness of this category of models for our task. Following Xu et al. (2015), we use the ResNet50 (He et al., 2016) as the image encoder and a unidirectional LSTM (Hochreiter and Schmidhuber, 1997) as the decoder for text. As the pretrained ResNet50 model is trained on object detection tasks on ImageNet (Deng et al., 2009), directly applying it to chart images gave poor results in our experiments. Also, we do not have any object labels for the chart images to train the encoder. Hence, we employ the recently proposed self-supervised strategy called *Barlow Twins* (Zbontar et al., 2021) which tries to make the embedding vectors of distorted versions of an image sample to be similar, while minimizing the redundancy between the components of these vectors. It achieves state-of-the-art results for ImageNet classification with an accuracy gap of only 3.3% from the supervised model. We pretrain a

[3] Our categorization of content is inspired by a recent study (Lundgard and Satyanarayan, 2022).

separate ResNet50 with Barlow Twins for each of our datasets and use it as an encoder in the model.

4.2 Data-to-text Models

• **Chart2text** (Obeid and Hoque, 2020) is an adapted transformer model for chart-to-text based on the data-to-text model of Gong et al. (2019). It takes a sequence of data records as input with each record being a set of tuples (*e.g.,* column header, cell value, column index) and embeds them into feature vectors with positional encodings to distinguish orders (Fig. 3a). The model includes an auxiliary training objective (binary labels indicating the presence of the record in the output sequence) on the encoder to maximize the content selection score. It also implements a templating strategy of target text with data variables (*e.g., cells*, *axis labels*) to alleviate hallucination problems. Since in Pew data tables are not available, we use OCR-generated texts as inputs which are linearized and embedded into feature vectors. The bounding box information of OCR-generated data of each chart is also embedded and concatenated to the table vectors to provide positional information to the model.

• **Field-Infusing Model** (Chen et al., 2020a) is inspired by the concept-to-text work (Lebret et al., 2016). The values in a cell are first encoded with an LSTM, which is then concatenated with the embeddings of row index and column heading. These table representations (h_1, h_2 in Fig. 3b) are then fed into a 3-layer Transformer encoder-decoder model to generate the target summaries. Additionally, for Pew, we embed the bounding box information of the chart OCR-texts and concatenate it to the LSTM-based field representation as an auxiliary positional information to the model.

• **BART** (Lewis et al., 2020) adopts a seq2seq Transformer architecture with denoising pretraining objectives. It is particularly pretrained to be effective for text generation tasks. For our chart-to-text tasks, we flatten the data table row by row and concatenate the title with table content as the input to the encoder (Fig. 3c). In the absence of data tables, we concatenate all the OCR-texts in a top to bottom order and fed it to the model as input.

• **T5** (Raffel et al., 2020) is a unified seq2seq Transformer model that converts various NLP tasks into a text2text generation format. It is first pretrained with a 'fill-in-the-blank' denoising objective, where 15% of the input tokens are randomly dropped out. The spans of consecutive dropped-out tokens are replaced by a sentinel token. The decoder then has to predict all of the dropped-out token spans, delimited by the same sentinel tokens used in the input. This is different from the pretraining objective of BART where the decoder predicts the entire original sequence (not just the dropped spans). T5 is fine-tuned with several supervised multi-task training objectives (*e.g.,* machine translation, text summarization). We format the input in the same way as for the BART models. Specifically, we add "translate Chart to Text: " to the prefix of the input to mimic the pretraining process (see Fig. 3c).

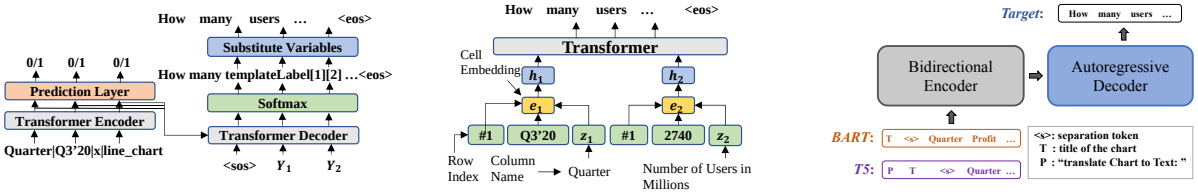
For OCR-based input, we experiment with two T5 model variants. In the first variant, we concatenate all the OCR-extracted sentences from the chart image in a top to bottom order and fed it to the model as input. In the second, we modify the input to accommodate the spatial information of the detected texts. Inspired by Tan and Bansal (2019), we feed the bounding box coordinates of each detected text token into a linear layer to produce positional embeddings which are then added to their corresponding embeddings of the OCR tokens as input.

5 Evaluation

5.1 Automatic Evaluation

Measures For automatic evaluation of the summary quality, we utilized five measures. BLEU (Post, 2018) and CIDEr (Vedantam et al., 2015) measure n-gram overlaps between the model generated text and the reference text. CIDEr computes TF-IDF weighted n-gram overlaps. BLEURT (Sellam et al., 2020) is a model-based evaluation metric that indicates to what extent the candidate is grammatical and conveys the meaning of the reference. We use BLEURT-base-128. Content Selection (CS) metric measures how well the generated summaries match the gold summaries in terms of selecting records to generate (Wiseman et al., 2017). Since both the BLEURT and CS are calculated at the sentence-level, we average these scores over the whole test set. Finally, for readability and fluency, we measure Perplexity (PPL) using a pre-trained GPT-2 Medium (Radford et al., 2019).

Results In general, from the results in Table 4, we notice that large-scale unsupervised pretraining (*i.e.,* " -BART", " -T5") helps to boost the performance significantly. In terms of the model variants, the image captioning model has failed to capture



(a) Chart2text model (b) Field-infusing model (c) BART/T5 fine-tuning

Figure 3: Different chart2text model architectures. Fig. 3c shows fine-tuning stage of the training (not unsupervised pretraining)

Models	BLEU ↑	CS ↑	BLEURT ↑	CIDEr ↑	PPL ↓
		Statista			
Image Caption	15.94	25.70%	-0.76	0.95	10.53
TAB-Chart2text	21.10	56.10%	0.06	2.61	28.79
TAB-Field-Infuse	12.09	42.07%	-0.32	1.78	17.01
TAB-BART	36.36	**77.14%**	0.12	4.40	12.55
TAB-T5	**37.01**	75.72%	**0.15**	**4.68**	10.00
OCR-T5	35.29	73.77%	0.10	4.43	8.66
OCR-T5*	34.55	73.55%	0.09	4.37	**8.59**
TAB_OCR-Chart2text	7.64	47.58%	-0.44	1.09	54.98
TAB_OCR-Field-Infuse	7.03	37.63%	-0.49	1.18	14.76
TAB_OCR-BART	35.83	72.15%	0.09	3.97	13.99
TAB_OCR-T5	36.74	72.22%	0.13	4.33	10.20
		Pew			
Image Caption	4.09	2.14%	-0.96	0.38	16.43
OCR-Chart2Text*	7.20	24.49%	-0.56	0.65	12.11
OCR-Field-Infuse*	0.19	10.12%	-1.01	0.26	9.57
OCR-BART	9.09	39.99%	-0.38	1.97	11.04
OCR-T5	**10.49**	**40.87%**	**-0.35**	**2.20**	10.11
OCR-T5*	10.42	40.31%	-0.42	2.13	**8.65**

Table 4: Evaluation results for different models on Statista and Pew test sets. ↑ : Higher is better, ↓ : Lower is better. "TAB- " models have access to the underlying data table and "OCR- " models use OCR-extracted data. OCR variants with ⋆ superscript use bounding box information. "TAB_OCR- " models use automatically generated data tables.

relevant information from charts (low CS score) even though it generates fluent text (low PPL).

On Statista, when the data tables are available, Chart2text and Field-Infuse models are able to extract information from the data table, but they struggle to produce texts with good quality. This could be because these models did not use any large-scale pretraining. On the other hand, TAB-BART and TAB-T5 are able to produce well-structured and relevant summaries. The OCR-based models can generally generate fluent summaries but they are slightly less effective in extracting the relevant information since the OCR process introduces some noise in the input data.

We also experiment with automatically extracted tables to see how the models perform in the absence of gold data tables. To this end, we extended ChartOCR (Luo et al., 2021), which predicts the raw data values of chart elements, to extract the fully-structured data table. The accuracy of automatic data extraction was 77.31% (see Appendix A.5 for details). We find that similar to OCR-based models, TAB_OCR-based models tend to be less effective in extracting the relevant information compared to their TAB-based counterparts which use ground truth data tables.

Pew, on the other hand, is much challenging because it contains many charts with ill-defined structure and the underlying data tables are not available. Unsurprisingly, the performance of all the models has dropped significantly compared to that on Statista. Nonetheless, we can see that without the presence of the underlying data table, the vision+text (OCR-based) models have brought notable improvements over the vision only model. Further breakdown of model performance based on chart types is provided in Appendix A.4.2.

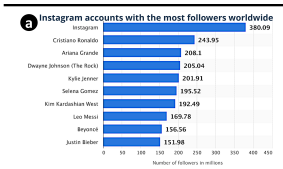
We also evaluate the *transferability* of the models and the datasets, where we first pretrain a model on a source dataset and fine-tune it on the target dataset. In addition to our two datasets (Statista or Pew), we experiment with ToTTo (Parikh et al., 2020) as another source dataset, which is a large-scale open-domain English table-to-text dataset. Our results show that pretraining on other datasets only brings about marginal improvement. Details of this experiment can be found in Appendix A.4.1.

5.2 Human Evaluation

To further assess the summary quality we performed a human evaluation on 150 randomly sampled charts from the Statista dataset with four internal annotators who are native speakers of English. For each chart, annotators performed pairwise comparisons between the outputs of TAB-T5, OCR-T5 and the original gold summary (served as a control), resulting in a total of 450 pairwise comparisons (Appendix A.4.3). They compared the summaries based on three criteria: (*i*) **Factual correctness**: Which summary is more factually

	TAB-T5 (1) vs. OCR-T5 (2)			Gold (1) vs. TAB-T5 (2)			Gold (1) vs. OCR-T5 (2)		
Summary	**Factual**	**Coherence**	**Fluency**	**Factual**	**Coherence**	**Fluency**	**Factual**	**Coherence**	**Fluency**
Summary 1 Win	55.3%	23.3%	20.0%	30.0%	36.7%	22.0%	59.3%	43.3%	28.7%
Summary 2 Win	12.0%	11.3%	11.3%	13.3%	16.7%	14.0%	7.33%	15.3%	17.3%
Tie	32.7%	65.3%	68.7%	56.7%	46.7%	64.0%	33.3%	41.3%	54.0%
p-value (sign test)	1.86e-11	8.77e-3	0.0395	1.31e-3	5.26e-4	0.0668	1.27e-16	4.25e-6	0.0266

Table 5: Human evaluation results for comparing between the outputs of TAB-T5, OCR-T5 and the gold summary.



Gold: Footballer Cristiano Ronaldo heads the ranking of the most popular Instagram accounts as of December 2020. He is the most-followed person on the photo sharing app platform with nearly 244 million followers. Instagram's own account was ranked first with over 369 million followers.

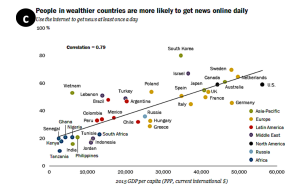
TAB-T5 As of December 2020, the Instagram account with the most followers worldwide was Cristiano Ronaldo's account with 243.95 million followers. The social network's most popular account was Instagram, which had 380.09 million followers.

OCR-T5: As of December 2020, Cristiano Ronaldo was the Instagram account with the most followers worldwide, with 380.09 million followers. The Brazilian singer had 243.95 million followers on the photo sharing app.

Gold: Germans give the U.S. some of its worst ratings in the survey. Only 26% have a positive view of America, while just 10% have confidence in Trump when it comes to his handling of world affairs. These opinions are in stark contrast to the very favorable assessments Germans had during Barack Obama's presidency, but roughly on par with views at the end of George W. Bush's tenure.

OCR-BART Roughly nine-in-ten Germans (88%) express a favorable view of the U.S. This is up sharply from 73% in 2016, but it also is the highest percentage since Pew Research Center began surveying in Germany in 2000. Germans' views of America have also become more positive since the end of the George W. Bush era..

OCR-T5 Germany is a country where U.S. image has declined sharply in recent years. For example, in 2008, before the coronavirus outbreak began to spread, about a third of Germans had confidence in President Barack Obama to do the right thing in world affairs. By 2014, confidence in Obama had fallen to about a third, but by 2019 confidence had nearly doubled to about a quarter.



Gold: In general, internet access has been shown to be higher in wealthier countries, and this plays out to a greater likelihood of using the internet for news as well. For example, 61% in Australia – which had a 2015 gross domestic product (GDP) per capita of $46,271 – get news at least once a day through the internet. Just 20% in Senegal, with a GDP per capita of $2,421, do the same.

OCR-BART People in wealthier countries are more likely than those in poorer nations to get news online at least once a day. This is especially the case in Venezuela, where about seven-in-ten people (71%) get news daily.

OCR-T5 However, it is not the case that the correlation between digital news consumption and personal income is not quite so strong. The study found that in wealthier countries, people were more likely to use the internet for news on a daily basis than in poorer countries. The finding that people in wealthier countries tended to do this more often than those in poorer nations to get news online.

Figure 4: Sample outputs from Statista (first column) and Pew datasets (last two columns). Red indicates hallucination errors and blue indicates tokens that are resulted in factual errors in the model output.

correct (*i.e.,* facts mentioned are supported by the chart)? (*ii*) **Coherence**: Which summary is more coherent (*i.e.,* sentences are well connected)? and (*iii*) **Fluency**: Which summary is more fluent and grammatically correct? For each criterion, the annotator picked the better one (win) or equally good (tie). Each comparison was performed by one annotator, except the first 150 comparisons for which we had two annotators to measure the agreement. The agreement for these 150 comparisons, excluding ties, was 74.3% (ties were excluded since they do not affect the overall ranking of the summaries).

Table 5 shows that the TAB-T5 performed significantly better than OCR-T5 based on all three criteria, especially on factual correctness. This is likely because, without the data table as input, OCR-T5 model often fails to generate factually correct statements from the OCR text. We also observe that while the fluency of the model outputs is comparable to the gold summary, their factual correctness and coherence were significantly worse, especially for the OCR-T5 model.

5.3 Error Analysis and Challenges

We manually analyzed 200 random samples from Statista and Pew. We chose TAB-T5 and OCR-T5 for Statista and OCR-BART and OCR-T5 models for Pew. This analysis helps us to understand model errors and identify key challenges that existing models face as we describe below.

Perceptual and reasoning aspects As mentioned in §1, charts often describe complex patterns and trends which can be perceived by humans easily but they are not necessarily easy to derive through analysis of raw data tables. In Fig. 4b, the OCR-T5 model manages to describe a trend correctly in the first sentence but describes a trend incorrectly in the last sentence. These examples demonstrate the shortcomings of existing models. In order to explain perceptual and reasoning aspects effectively, we need more sophisticated models that better capture prominent visual relationships in charts. In particular, we aim to develop better representations including semantic graph representation of the chart that encodes numerical and logical relationships among chart objects.

Hallucinations Sometimes, the model outputs tokens that are irrelevant to the chart. For example, while the model outputs in Fig. 4a,b are quite fluent, they contain hallucination errors. This problem is commonly observed in other data-to-text work as well (Wiseman et al., 2017; Parikh et al., 2020).

Factual errors Factually incorrect statements are more common for the OCR-based models (*e.g.,* in Fig. 4a-b) since they do not take the data table as input, thus fail to associate the data values correctly. In contrast, TAB-T5 which utilizes the data table as input tends to generate less factual errors. This confirms that summarizing charts when the data table is not available is usually more challenging.

Computer vision challenges The factual errors illustrate some unique computer vision challenges. First, charts do not always show data values as text labels, thus the OCR models cannot access those values. Even if the data values are labeled, the absence of association between data values (*e.g.,* `Instagram` is related to `380.09M` in Fig. 4a) leads to factual errors. This problem might be alleviated if the model can extract the data table from a chart image. While there are some initial attempts in this direction (*e.g.,* Luo et al. (2021); Choi et al. (2019)), more accurate data extraction from charts is necessary.

Generalizability The charts in our benchmark cover several different chart types and a wide variety of topics (fig. 9). The charts in the Pew in particular have a wide variety of visual styles in terms of color, layout and typography as they were created over several years by different authors (see examples in fig. 1). Nevertheless, finding more chart-summary pairs with more diverse visual styles is an open challenge. In future, we aim to find more different sources of chart-summaries and perform cross-domain experiments across those different sources to evaluate the generalizability of models.

6 Conclusion

We have presented two large-scale datasets for chart summarization. We also provided several state-of-the-art baselines and measures. Our evaluation highlights the promise of these baselines and also reveals several unique challenges for the chart summarization task. We hope that Chart-to-text will serve as a useful research benchmark for model and metric development and motivate other researchers to explore this relatively new area.

Acknowledgement

The authors would like to thank the anonymous reviewers for their helpful comments. This research was supported by the Natural Sciences & Engineering Research Council (NSERC) of Canada.

Ethical Considerations

During the dataset collection and annotation process, we had many ethical issues to take into consideration. To respect the intellectual property of the chart publishers, we only used publicly available charts from resources that provide publication rights of downloaded content for academic purposes. According to the terms of use and publication rights for Statista,[4] users are granted publication rights only to free studies of Statista, so we only used the free publicly available webpages. According to the terms and conditions for Pew,[5] users are allowed to use the content as long as they are attributed to the Center or are not attributed to a different party.

To fairly compensate the Mechanical Turk annotators, we compensated the annotators based on the minimum wage in the United States at the time (7.25 US$ per hour) and the estimated time taken for each task (1 minute). Hence, these annotators received 0.10 - 0.15 US$ for each chart, depending on the number of candidate paragraphs associated with it. Additionally, to protect the privacy of these annotators, all of their annotations were anonymized.

To ensure the reproducibility of our experimental results, we have provided the hyperparameter settings and estimated training time in Appendix A.3.

We foresee one possible misuse of our models that is to spread misinformation. Currently, our model outputs tend to appear fluent but contain some hallucinations and factual errors, as detailed in §5.3. Hence, if such model outputs are published without being corrected, it may mislead and misinform the general public.

References

2021. Wikidata knowledge base.

Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi

[4] https://www.statista.com/getting-started/publishing-statista-content-terms-of-use-and-publication-rights

[5] https://www.pewresearch.org/about/terms-and-conditions/

Parikh, Stefan Lee, and Peter Anderson. 2019. nocaps: novel object captioning at scale. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8948–8957.

Jeonghun Baek, Geewook Kim, Junyeop Lee, Sungrae Park, Dongyoon Han, Sangdoo Yun, Seong Joon Oh, and Hwalsuk Lee. 2019a. What is wrong with scene text recognition model comparisons? dataset and model analysis. In *International Conference on Computer Vision (ICCV)*.

Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. 2019b. Character region awareness for text detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9365–9374.

Regina Barzilay and Mirella Lapata. 2005. Collective content selection for concept-to-text generation. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 331–338, Vancouver, British Columbia, Canada. Association for Computational Linguistics.

Sandra Carberry, Stephanie Elzer, and Seniz Demir. 2006. Information graphics: an untapped resource for digital libraries. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 581–588.

Charles Chen, Ruiyi Zhang, Eunyee Koh, Sungchul Kim, Scott Cohen, Tong Yu, Ryan A. Rossi, and Razvan C. Bunescu. 2019. Figure captioning with reasoning and sequence-level training. *CoRR*, abs/1906.02850.

Wenhu Chen, Jianshu Chen, Yu Su, Zhiyu Chen, and William Yang Wang. 2020a. Logical natural language generation from open-domain tables. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7929–7942, Online. Association for Computational Linguistics.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollar, and C. Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server.

Zhiyu Chen, Wenhu Chen, Hanwen Zha, Xiyou Zhou, Yunkai Zhang, Sairam Sundaresan, and William Yang Wang. 2020b. Logic2Text: High-fidelity natural language generation from logical forms. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2096–2111, Online. Association for Computational Linguistics.

J. Choi, Sanghun Jung, Deok Gun Park, J. Choo, and N. Elmqvist. 2019. Visualizing for the non-visual: Enabling the visually impaired to use visualization. *Computer Graphics Forum*, 38.

Zhe Cui, Sriram Karthik Badam, M Adil Yalçin, and Niklas Elmqvist. 2019. Datasite: Proactive visual data exploration with computation of insight-based recommendations. *Information Visualization*, 18(2):251–267.

Seniz Demir, Sandra Carberry, and Kathleen F. McCoy. 2012. Summarizing information graphics textually. *Computational Linguistics*, 38(3):527–574.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Massimo Fasciano and Guy Lapalme. 1996. Postgraphe: a system for the generation of statistical graphics and text. In *Eighth International Natural Language Generation Workshop*.

Leo Ferres, Gitte Lindgaard, Livia Sumegi, and Bruce Tsuji. 2013. Evaluating a tool for improving accessibility to charts and graphs. *ACM Trans. Comput.-Hum. Interact.*, 20(5).

Li Gong, Josep Crego, and Jean Senellart. 2019. Enhanced transformer model for data-to-text generation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 148–156, Hong Kong. Association for Computational Linguistics.

Nancy L Green, Giuseppe Carenini, Stephan Kerpedjiev, Joe Mattis, Johanna D Moore, and Steven F Roth. 2004. Autobrief: an experimental system for the automatic generation of briefings in integrated text and information graphics. *International Journal of Human-Computer Studies*, 61(1):32–70.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Ting-Yao E. Hsu, C. Lee Giles, and Ting-Hao K. Huang. 2021. Scicap: Generating captions for scientific figures. In *Findings of 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021 Findings)*.

Dae Hyun Kim, Vidya Setlur, and Maneesh Agrawala. 2021. Towards understanding how readers integrate charts and captions: A case study with line charts. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–11.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1203–1213, Austin, Texas. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Zhuo Li, Matthew Stagitis, Sandra Carberry, and Kathleen F McCoy. 2013. Towards retrieving relevant information graphics. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 789–792.

Alan Lundgard and Arvind Satyanarayan. 2022. Accessible Visualization via Natural Language Descriptions: A Four-Level Model of Semantic Content. *IEEE Trans. Visualization & Comp. Graphics (Proc. IEEE VIS)*.

Junyu Luo, Zekun Li, Jinpeng Wang, and Chin-Yew Lin. 2021. Chartocr: Data extraction from charts images via a deep hybrid framework. *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1916–1924.

Hongyuan Mei, TTI UChicago, Mohit Bansal, and Matthew R Walter. 2016. What to talk about and how? selective generation using lstms with coarse-to-fine alignment. In *Proceedings of NAACL-HLT*, pages 720–730.

Vibhu O. Mittal, Johanna D. Moore, Giuseppe Carenini, and Steven Roth. 1998. Describing complex charts in natural language: A caption generation system. *Computational Linguistics*, 24(3):431–467.

Jason Obeid and Enamul Hoque. 2020. Chart-to-text: Generating natural language descriptions for charts by adapting the transformer model. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 138–147. Association for Computational Linguistics.

Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. Totto: A controlled table-to-text generation dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1173–1186.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Mª del Puy Pérez-Echeverría, Yolanda Postigo, and Cristina Marín. 2018. Understanding of graphs in social science undergraduate students: selection and interpretation of graphs. *Irish Educational Studies*, 37(1):89–111.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *Open-AI Blog*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Ehud Reiter. 2007. An architecture for data-to-text systems. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, pages 97–104. Association for Computational Linguistics.

Ehud Reiter, Somayajulu Sripada, Jim Hunter, Jin Yu, and Ian Davy. 2005. Choosing words in computer-generated weather forecasts. *Artificial Intelligence*, 167(1-2):137–169.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. *arXiv preprint arXiv:2004.04696*.

Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. 2020. Textcaps: a dataset for image captioning with reading comprehension.

Andrea Spreafico and Giuseppe Carenini. 2020. Neural data-driven captioning of time-series line charts. In *Proceedings of the International Conference on Advanced Visual Interfaces*, AVI '20, New York, NY, USA. Association for Computing Machinery.

Arjun Srinivasan, Steven M Drucker, Alex Endert, and John Stasko. 2018. Augmenting visualizations with interactive data facts to facilitate interpretation and communication. *IEEE transactions on visualization and computer graphics*, 25(1):672–681.

Yixuan Su, David Vandyke, Sihui Wang, Yimai Fang, and Nigel Collier. 2021. Plan-then-generate: Controlled data-to-text generation via planning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*. Association for Computational Linguistics.

Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. 2021. Training data-efficient image transformers & distillation through attention. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10347–10357. PMLR.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575.

Douglas Whitaker and Tim Jacobbe. 2017. Students' understanding of bar graphs and histograms: Results from the locus assessments. *Journal of Statistics Education*, 25(2):90–102.

Sam Wiseman, Stuart M Shieber, and Alexander M Rush. 2017. Challenges in data-to-document generation. *arXiv preprint arXiv:1707.08052*.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*.

Zichao Yang, Phil Blunsom, Chris Dyer, and Wang Ling. 2017. Reference-aware language models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1850–1859.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. 2021. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. 2021. Vinvl: Making visual representations matter in vision-language models. *CVPR 2021*.

A Appendices

A.1 Additional Details on Data Annotation

A.1.1 Example Webpage from Statista

An example of a webpage from Statista is given in Fig. 5. It contains a chart image and its accompanying description text. The first part of the text (highlighted in blue) provides a succinct summary of the chart while the remaining parts of the text (not highlighted) provides irrelevant background information, such as Facebook's history.

A.1.2 Annotation of x-axis Labels in Statista

The user interface for the annotation task of labeling the x-axis labels in the Statista dataset is given in Fig. 6.

A.1.3 Identify Candidate Paragraphs in Pew

The details for computing the relevance score of a paragraph to the given chart, and the heuristic for finding relevant paragraphs in the Pew dataset are given in Fig. 7.

A.1.4 Relevant Paragraph Selection in Pew

For the relevant paragraph selection task, the annotators received 0.10 - 0.15 US$ for each chart, depending on the number of candidate paragraphs associated with it. To ensure the quality, we recruited participants with at least 95% approval rate and 5000 approved HITs (Human Intelligence Tasks) and they were only allowed to complete the tasks after they successfully completed a sample task.

The user interface for the Mechanical Turk annotation task of selecting paragraphs relevant to charts in the Pew dataset is given in Fig. 8.

A.2 Dataset Analysis

Figure 9 shows the distribution of topics in two datasets.

A.3 Chart-to-text Baseline Models

The experiments are done on our machine (CPU: Intel(R) Xeon(R) Gold 6240 CPU @ 2.60GHz, GPU: 4 × NVIDIA GTX 2080Ti). Training T5 is the most computationally costly task, which takes around 16-20 hours on 4× GPUs.

Image Captioning Models For pretraining the image encoders and captioning model, we follow the same training setup as presented in the original papers. Inference is done with beam search with a beam size of 4.

Figure 5: A screenshot of a webpage from Statista.

```
2

                      United States: estimated net worth of the 20 richest people as of March 2020

                          (in billion U.S. dollars)

 This statistic presents the estimated net worth of the 20 richest people in America as of March 2020. At this time Jeff Bezos was the richest man in
  the United States with an estimated net worth of 113 billion U.S. dollars.
0                   Jeff Bezos (Amazon)
1                Bill Gates (Microsoft)
2     Warren Buffet (Berkshire Hathaway)
3                Larry Ellison (Oracle)
4            Mark Zuckerberg (Facebook)
5                  Jim Walton (Walmart)
6                Alice Walton (Walmart)
7       S. Robson "Rob" Walton (Walmart)
8              Steve Ballmer (Microsoft)
9                   Larry Page (Google)
10                Sergey Brin (Google)
11      Michael Bloomberg (Bloomberg LP)
12        Charles Koch (Koch Industries)
13          Julia Koch (Koch Industries)
14              MacKenzie Bezos (Amazon)
15                   Phil Knight (Nike)
16     Sheldon Adelson (Las Vegas Sands)
17                    John Mars (Mars)
18               Jacqueline Mars (Mars)
19       Elon Musk (Tesla Motors, SpaceX)
Name: human, dtype: object
keep it or change(0keep,1change): 1
the new column name: Richest people
```

Figure 6: The user interface for labeling the x-axis labels in the Statista dataset.

Let s_i be the relevance score for sentence i in the paragraph.
Let l_i be the number of lexical token matches between sentence i and the chart.
Let n_i be the number of numerical token matches, excluding year tokens, between sentence i and the chart.
Let y_i be the number of year token matches between sentence i and the chart.
Let u_i be the number of numerical tokens that appear in sentence i but not in the chart.
Let c be the number of sentences in the paragraph.
$s_i = 0.58l_i + 1.4n_i - 0.5u_i$

Let $content$ be the content score of the paragraph.

$$content = \frac{1}{1 + \exp\left(0.3 \times \left(-\max_i\left(s_i\right) + 1.7\right)\right)}$$

Let $proximity$ be the proximity score of the paragraph.
Let $dist$ be the proximity of the paragraph to the chart. $-5 \leq dist \leq 5$
For example, $dist = -1$ if the paragraph is directly before the chart, $dist = 0$ if it contains the chart and $dist = 1$ if it is directly after the chart.
$proximity = 0.4 \times \exp\left(-0.1 \times |dist|^2\right) + 0.6$

Let rel be the relevance score of the paragraph.
$rel = content \times proximity$

Heuristic: A paragraph is relevant if it satisfies the following conditions:

$$\sum_i l_i > 3$$

$$\sum_i n_i + y_i > 0$$

$$\sum_i u_i = 0$$

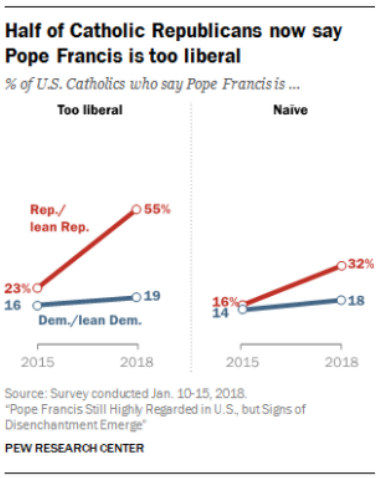
$$rel > 0.72$$

$$c > 0$$

Figure 7: The computation of a paragraph's relevance score to a chart, and the conditions for the heuristic in the Pew dataset.

View instructions

Chart



Paragraphs

For the given chart image, please evaluate whether each paragraph is completely irrelevant, partially relevant or completely relevant to the chart. Please click "View Instructions" to view the detailed criteria for determining whether a paragraph is relevant to a chart and to see some sample answers. Below is a summary of the criteria for determining whether a paragraph is relevant to a chart.

Relevant Information

- Data that can be found or computed from the chart
- Trends that can be derived from the chart (e.g. increasing, decreasing)

Irrelevant Information

- Data that is related to the chart, but cannot be found or computed from the chart
- Background information (e.g. past results, methodology, historical information)

Paragraph #1

A growing share of Catholics see the pope as being too liberal (34%) as well as naïve (24%), up from 19% and 15%, respectively, in 2015. This is especially true among Catholics who are Republican or who lean Republican; they are much more likely to see Francis as being too liberal than Democrats or those who lean Democratic (55% vs. 19%). A partisan gap also exists among Catholics on views about whether the pope is naïve (32% among Republicans vs. 18% among Democrats).

How relevant is paragraph #1 to the chart?

○ Completely Irrelevant ○ Partially Relevant ○ Completely Relevant

○ Can't Decide (Please justify below)

Brief justification for your answer (if you can't decide)

Submit

Figure 8: The user interface for the Mechanical Turk annotation task in the Pew dataset.

Pre-train Dataset	Fine-tune Dataset	BLEU
Totto	Pew	10.66
Totto	Statista	37.19
Pew	Statista	37.32
Statista	Pew	10.73

Table 6: Results measured by BLEU for transferability based on the T5 model.

Chart2text We follow the same settings of Obeid and Hoque (2020) with 1 encoder layer, 6 decoder layers and a dropout ratio of 0.1, and train the model for 80 epochs with a batch size of 6. For inference, we use beam search with a beam size of 4.

Field-Infusing Model We follow the same settings as Chen et al. (2020a) and train the model for 10 epochs with a dropout ratio of 0.1 and batch size of 1.

BART We fine-tune BART-Base[6] (140M, 6-layers) for 500K iterations with a batch size of 4 and evaluate after every 2,000 iterations on the validation set. The initial learning rate is set to 0.0005. For inference, we use the model with the lowest validation loss and decode with a beam size of 4.

T5 Similar to BART, we fine-tune T5-Base[6] (220M, 12-layer Transformer as the encoder and decoder) for 500K iteration with a batch size of 4 and an initial learning rate of 0.0005, evaluate after every 2,000 iterations on validation set, and use the model with best validation loss for testing. Inference is done with beam search with a beam size 4.

A.4 Additional Results from Evaluation

A.4.1 Transfer Results

Since both Statista and Pew share some of the topics with each other, we conduct transfer experiment to verify if pretraining on one dataset could help to improve the final results on the other. In addition, since table-to-text has similarities with our task, we also experiment with pretraining on a large-scale open-domain English table-to-text dataset ToTTo (Parikh et al., 2020) before training on our datasets. We use full table for ToTTo since our task does not contain highlighted cell. Pretraining and fine-tuning use T5-based models and have the same training procedure as described in §4.2. From Table 6, we see that pretraining on other datasets only improves the final performance by a small margin.

[6] huggingface.co/transformers

A.4.2 Performance by Chart Types

Chart Types	Bar	Line	Pie	Table
BLEU	36.46	45.28	21.35	26.12
PPL	10.08	7.53	8.79	11.34
CIDEr	4.62	5.59	3.27	3.67
BLEURT	0.14	0.27	-0.13	-0.22

Table 7: Results on Statista test set *w.r.t.* chart types.

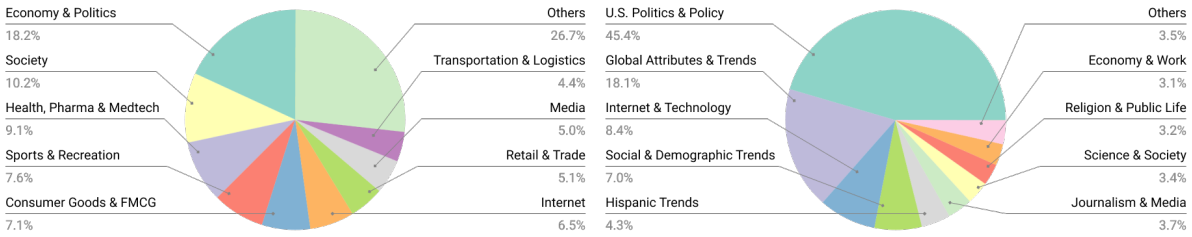
Chart types can influence the performance of the model. We present the performance breakdown on Statista of our best model (*i.e.,* TAB-T5) based on chart types in Table 7. We observe that the model is good at summarizing simple and frequent chart types (*e.g.,* line chart), whereas the model is less effective in generating informative summaries for complex and less frequent charts (*e.g.,* pie charts) in our datasets.

A.4.3 Human Evaluation

The user interface for the human evaluation annotation task of comparing chart summaries is given in Fig. 10.

A.5 Automatic Data Extraction from Charts

Model: We extend ChartOCR (Luo et al., 2021) which combines deep-learning and rule-based methods to extract the underlying data values from the chart images. First, key-point detection networks detects the chart main elements (e.g. plot area, y-axis-title, x-axis-title, and legend area) and marks (e.g. bars, line points, and pie slices). We extend the detection network to detect textual labels and the legend marks in the chart (see an example in Figure 11). For the rectangular objects, the network outputs the top-left and bottom-right points which are grouped together based on the distance. For lines, the network outputs the coordinates of the line points which are grouped together based on the color. For pie charts, the network outputs the separating points between the slices along the perimeter of the pie. As shown in Figure 11, the scale of the chart is estimated using the *y-axis-labels'* values and y coordinates. Finally, the data values of the chart marks (e.g. bars, line points) are calculated using the scale of the chart. For pie charts, the values are estimated by calculating the angle between each two neighbouring points.



(a) Statista (b) Pew

Figure 9: Distribution of topics in the two datasets.

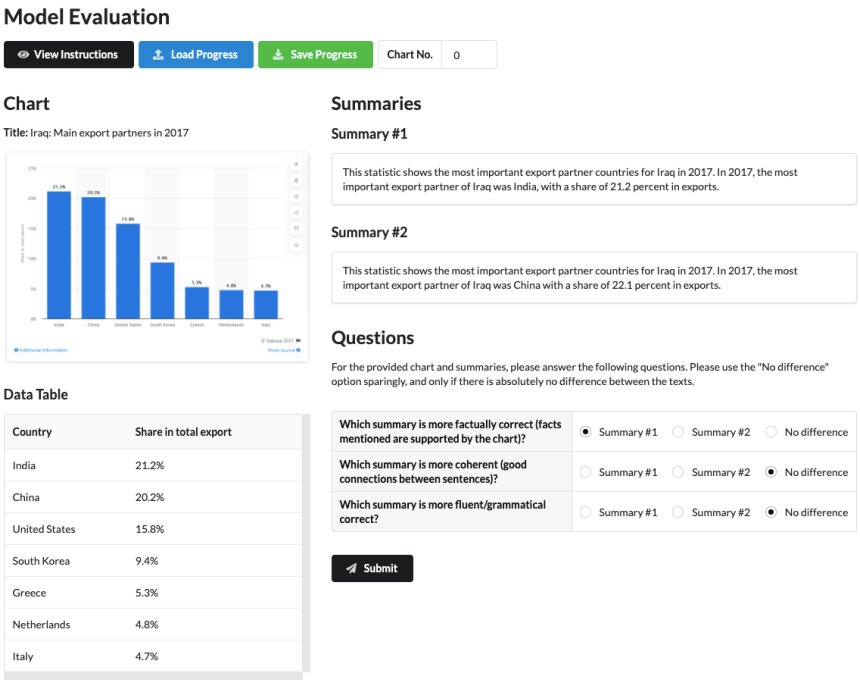


Figure 10: The user interface for human evaluation: it presents two summaries at a time and asks the participant to compare between them based on three measures.

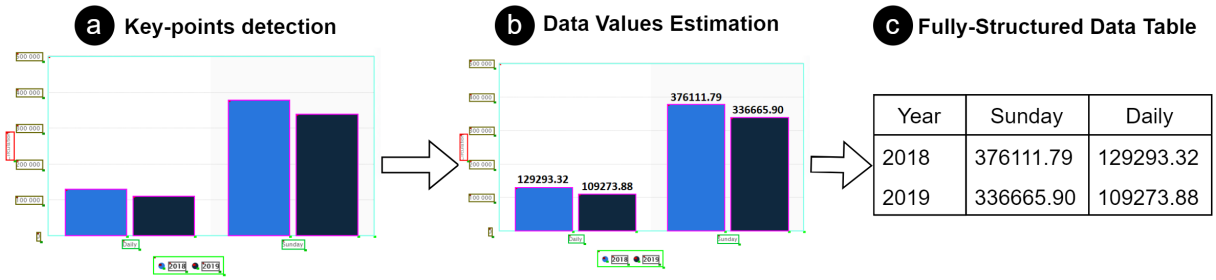


Figure 11: Data Extraction example from Statista.

Since the original ChartOCR model only outputs the raw data values, we we further extend their approach to output the fully-structured data table as follows. First, we utilize the CRAFT model (Baek et al., 2019a) to recognize the texts of the detected textual chart elements (*x-axis labels*, and *legend labels*). Then, we associate the data values with their closest *x-axis-label* and the data series (e.g. a group of bars or line points) with the legend labels based on the color. For example, in Figure 11b, the bars are matched with their closest *x-axis-labels* ('Sunday' and 'Daily'). Moreover, the values of dark blue bars are associated with '2019' *legend-label* and the values of light blue bars are associated with

'2018' *legend-label* based on the matched colors. In this way, our approach recovers the fully structured data table from the chart as shown in Figure 11c.

Evaluation Metric: We evaluate our extracted data table using the following metric (adapted from ChartOCR (Luo et al., 2021)). We define the distance function between two data points as:

$$D(gt, pr) = min(1, ||\frac{gt - pr}{gt}||)$$

where gt is the ground truth value and pr is the predicted value. We then compute the cost matrix C, where $C_{n,m} = D(gt_n, pr_m)$. The total minimum cost is then estimated by solving the linear sum assignment problem as follows:

$$cost = \sum_{i=1}^{K} \sum_{j=1}^{K} C_{i,j} X_{i,j}$$

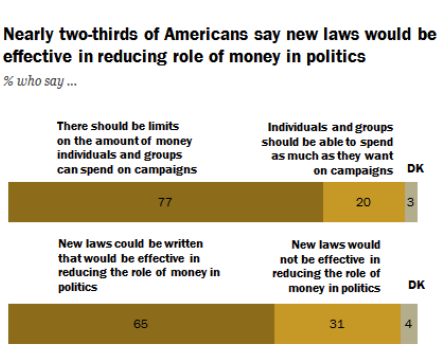
Where $K = max(N, M)$ and X is a binary assignment matrix. The final score is then computed using the following equation:

$$score = 1 - \frac{cost}{K}$$

Finally, we average the scores of all the charts to compute the overall score.

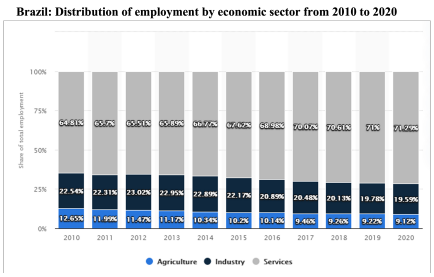
A.6 Additional Examples from Statista and Pew datasets

Figure 12 presents additional samples from our chart-to-text benchmark covering a diverse range of chart types and styles.

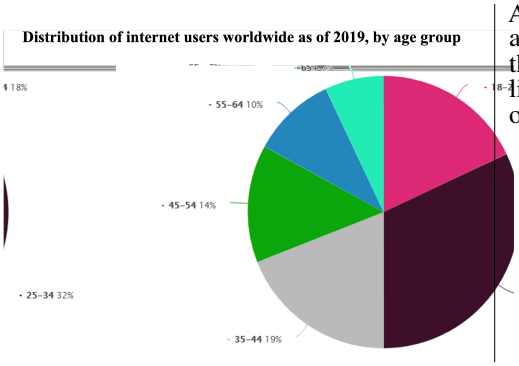


Americans overwhelmingly support limits on political campaign spending, and most think new laws could effectively reduce the role of money in politics. And there is extensive support for reining in campaign spending: 77% of the public says "there should be limits on the amount of money individuals and organizations" can spend on political campaigns; just 20% say they should be able to spend as much as they want. A somewhat smaller majority (65%) says that new campaign finance laws could be written that would be effective in reducing the role of money in politics, while 31% say any new laws would not be effective.

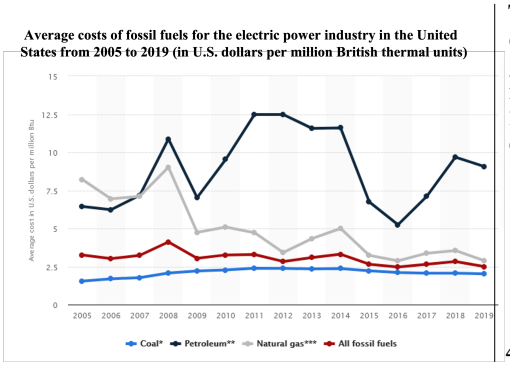
In a recent survey of what Americans know about science, we asked people to interpret the chart you see here and tell us what it showed. Six-in-ten (63%) identify the best interpretation of this chart as "the more sugar people eat, the more likely they are to get cavities."



The statistic shows the distribution of employment in Brazil by economic sector from 2010 to 2020. In 2020, 9.12 percent of the employees in Brazil were active in the agricultural sector, 19.59 percent in industry and 71.29 percent in the service sector.



As of 2019, a third of online users worldwide were aged between 25 and 34 years. Website visitors in this age bracket constituted the biggest group of online users worldwide. Also, 18 percent of global online users were aged 18 to 24 years.



The cost of fossil fuels in the electric power industry can vary depending on the source that is used. In general, fossil fuels cost about 2.50 U.S. dollars per million British thermal units (Btu) but can range from 2.02 U.S. dollars per million Btu for coal to 9.07 U.S. dollars per million Btu for petroleum.

Figure 12: Examples of chart-summary pairs in our benchmark. The top two examples are from the Pew research dataset and the rest of the examples are from the Statista dataset.