LineCap: Line Charts for Data Visualization Captioning Models

Anita Mahinpei* Zona Kostic† Chris Tanner‡

Harvard University

ABSTRACT

Data visualization captions help readers understand the purpose of a visualization and are crucial for individuals with visual impairments. The prevalence of poor figure captions and the successful application of deep learning approaches to image captioning motivate the use of similar techniques for automated figure captioning. However, research in this field has been stunted by the lack of suitable datasets. We introduce *LineCap*, a novel figure captioning dataset of 3,528 figures, and we provide insights from curating this dataset and using end-to-end deep learning models for automated figure captioning.

Keywords: figure captioning, line charts, deep learning dataset

1 INTRODUCTION

Data visualizations are commonly used in scientific papers to convey complementary information and enhance readers' comprehension. Figure captions can help readers understand the purpose of a visualization, and they are often the only means for individuals with visual impairments to access figures. While guidelines for creating accessible visualizations exist, many authors have yet to adopt these practices when writing figure captions [22]. As such, automatic caption generation for data visualizations can significantly alleviate information inaccessibility for people with visual impairments.

Early efforts for automated figure captioning have primarily focused on developing rule-based and non-deep learning techniques with modular pipelines [1, 7, 9, 23, 24]. While these methods do not need large corpora of figures and captions, they are highly specialized and do not generalize well to other chart types and styles, or to charts that require more complex high-level insights.

The advancement of deep learning has led to significant improvements in the field of automated image captioning [20]. Researching novel neural network architectures for image caption generation would not have been possible without the curation of publicly-shared datasets used for training and evaluation of these models. These datasets tend to contain large amounts of images scraped from the web with several captions crowd-sourced for each image. The Flickr30K dataset [29], the Microsoft COCO dataset [21], and the VizWiz-Captions dataset [10] are just a few examples of many image captioning datasets that have been created over the past decade.

The recent successes of deep learning approaches in automated image captioning motivate the application of similar approaches to the task of automated figure captioning. However, research in this field has been stunted by the lack of suitable training and evaluation datasets. Therefore, we introduce *LineCap*, a dataset containing line charts scraped from scientific papers each accompanied with crowd-sourced natural language descriptions. We share our design choices and challenges while curating LineCap, to help inform future creators of figure captioning datasets. We also establish baseline performances on LineCap and provide insights toward using deep learning models for automated figure captioning.

*e-mail: amahinpei@g.harvard.edu

†e-mail: zonakostic@g.harvard.edu

‡e-mail: christanner@g.harvard.edu

2 RELATED WORK

Previous research has developed neural networks that generate natural language descriptions for charts when provided with the underlying data in tabular form [5, 13, 26, 28, 36]. However, these data-to-text models and their accompanying datasets are not suitable for generating captions when the underlying figure data are not available, as is the case with most figures in scientific papers. Although Obeid et al. [26] provide the original chart images in addition to data tables and captions, their data were crawled from Statista[1], which only uses a limited set of chart styles and color schemes.

Chen et al. [4] and Qian et al. [31] use deep learning approaches to generate figure captions from chart images; however, these works trained models using FigureQA [16] and DVQA [14], which are *synthetic* figure question answering datasets. To generate reference captions for model training and evaluation, these works used a set of templates to create captions based on the question-answer pairs in the figure question answering datasets. The use of synthetic charts and template-based reference captions drastically limits the complexity of these datasets; the captions in these datasets only convey low-level information (e.g., chart type, axis titles, and global extrema) rather than high-level insights or trends. While describing low-level details is also important to visually impaired individuals, most low-level details can be extracted using figure parsing [30, 34] or question answering models [15, 35] and incorporated into captions using natural language models. Furthermore, Lundgard et al. [22] suggests that automated data visualization captioning research should primarily focus on describing overall trends and statistics. They categorize figure caption information into four broad groups: level 1 (e.g., chart type, labels, and axis ranges), level 2 (e.g., descriptive statistics and extrema), level 3 (e.g., complex trends and exceptions), and level 4 (e.g., domain specific insights and explanations). Through a user study, they found that blind readers consistently rank level 2 and 3 information as most useful [22]. However, captions provided by Chen et al. [4] and Qian et al. [31] commonly fall under level 1 and occasionally level 2.

To address the limitations of synthetic data, the research community has introduced non-synthetic datasets including SciCap [11], ChaTa+ [33], and Chart-to-Text [17]. SciCap is a dataset of figures and captions extracted from computer science papers published on arXiv between 2010 and 2020. ChaTa+ is a much smaller dataset of only 1,640 figures and captions extracted from scientific articles on arXiv and The World Health Organization (WHO). Chart-to-Text is another large figure captioning dataset that extends Obeid et al.'s data-to-text dataset. Its figures and captions predominantly consist of bar charts and are extracted from Pew Research[2] and Statista. While more diverse than synthetic datasets, the type and amount of information provided in figure captions scraped from scientific papers and online sources can vary dramatically. Some captions only include limited, level 1 information which is insufficient for visually impaired readers; other captions include extraneous, level 4 information which no model or layperson can deduce without access to external knowledge or the content of the article.

[1] https://www.statista.com/

[2] https://www.pewresearch.org/

3 LINECAP DATASET

Unlike some previous works that focused on creating datasets of bar charts, we created a benchmark dataset of line charts, as they are the second-most commonly occurring type of charts in scientific publications (second to diagrams [2]) and have more complex trends and patterns (i.e., level 3 information). We created a collection of line charts by taking a random sample of line plots from the SciCap dataset. SciCap [11] used PDFFigures 2.0 [6] to extract figures from scientific papers, then used a pre-trained classifier to identify the figure types – with a reported accuracy of 86%. As such, we manually inspected all sampled line plots to remove incorrectly cropped or classified figures. We also removed figures that were illegible due to poor quality, along with any multi-lined figures that were missing line labels or legends. Furthermore, to make the caption generation task easier for human annotators, we limited our scope to figures with at most five lines.

3.1 Caption Collection

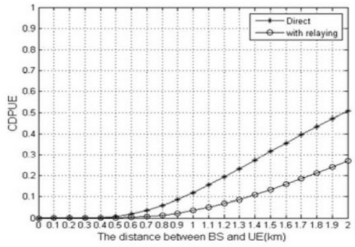
Although Kim et al. [19] provide guidelines for generating line chart captions, their guidelines are designed to enhance caption efficacy for sighted individuals. In this work, however, we focused on creating captions that enhance accessibility. As a result, following the recommendation of Lundgard et al. [22], we created captions that describe overall trends and statistics (i.e., level 2 and 3 captions). Level 3 information is perceiver dependent and cannot be generated from data tables, without reference to the visualization [22]. As such, we used Amazon Mechanical Turk to crowd-source captions, unlike some previous works that used data tables to fill out template captions. According to Morash et al. [25], when collecting chart descriptions from novice web workers, using query templates is more suitable than providing a set of written guidelines. We used a modified version of Morash et al.'s template for line chart descriptions. More specifically, since we were focusing on high-level information, we did not keep any parts of the template that extracted low-level details (e.g., axis and line labels). We also added a question to capture additional information such as notable comparisons between the lines in a multi-lined figure. This format provides the necessary structure to ensure annotators provide all the high-level details that must be included in an accessible caption. However, it does not limit sentence structures and the types of trends that could be described.

To ensure no low-level content from the chart is referenced in the high-level captions, annotators were instructed to number the lines based on the order in the figure legend, and to refer to the lines by their numbers (i.e., Line 1 to 5)[3]. Annotators were also instructed to use the term `xlabel` to refer to the x-axis label and `ylabel` to refer to the y-axis label. Our objective in excluding any references to labels from the figure is two-fold. First, labels in scientific figures tend to include subscripts, equations, or special characters that cannot be found in an English keyboard. By asking annotators not to use these labels in their descriptions, we avoid having to provide guidelines on transcribing these symbols. Second, this setup allows for training models that can be easily integrated with chart parsing models. Previous work on chart parsing can extract and classify text from figures [8, 30]. The classified texts can then replace the standardized terminology in our high-level captions.

Based on the results of an initial pilot study, we provided human annotators with a pictographic list of useful terminology for describing common line trends, along with video instructions that detail our Mechanical Turk interface. Because our task was writing intensive, we only accepted annotators from the following anglophone countries: Australia, Canada, New Zealand, UK, and USA. To ensure high quality, we only granted access to our annotation task to those who first passed a qualification test. We also regularly inspected random samples of chart descriptions from each annotator.

[3]Details can be found in our supplementary materials

The collected captions were processed to fix some spelling mistakes. We also followed a similar text normalization process to SciCap [11], whereby we replaced all references to axis values (e.g., 5.2, 10%, 100k) with the token `_value_`. We provide access to both the annotations with axis values and the normalized annotations[4].



SciCap Caption:
call dropping probability user equipment (ue) based performance vs. the distance between ue and base station (bs) . via amplify-and-forward moving relay the call dropping probability can be significantly decreased

Our Caption:
Number of Lines: 2
Line 1's Trend: the line starts out horizontal, then it increases at a constant rate.
Line 2's Trend: the line starts out horizontal, then it increases at a constant rate.
Overall Trend: Line 1 and 2 begin together with line 1 increasing to end on ylabel value above line 2.

Figure 1: A sample figure-annotation pair from LineCap compared to the caption provided in the SciCap dataset.

Table 1: Number and % of figures with the specified number of lines

number of lines	number of figures	% of total figures
1	570	16%
2	1025	29%
3	829	23%
4	796	23%
5	308	9%

3.2 Dataset Analysis

LineCap contains 3,528 figures, each with at least one human annotation that specifies: the number of lines in the figure; a separate description for the trend of each line in the figure; and, an overall chart description. Most figures contain only a single annotation, but some figures have up to three different annotations. Our resulting dataset has a total of 3,964 annotations. The distribution of the number of lines in each figure can be found in Table 1. Figures most often have 2 lines, while the least common figures have 5 lines. An average description for a line trend is 14 words long, while an average description for the overall chart is 26 words long. Most annotations (both for individual lines and the overall chart), do not make any references to axis values from the chart. After pre-processing the descriptions (i.e., removing stop words and lemmatizing), the 10 most common words are: line, increase, decrease, rate, ylabel, trend, xlabel, value, roughly, and constant. Excluding graph values, the pre-processed descriptions have a total of 925 unique words.

Individual line trends tend to describe the type of change (e.g., increase, decrease, constant) and the rate of change (e.g., increasing, decreasing or constant rate) of the line. Sometimes the trends also indicate the presence of noise, peaks, and troughs in the line. If a figure line is composed of multiple segments with different trends (e.g., first increasing then horizontal), descriptions tend to specify the trend of each segment in chronological order, but they do not indicate where exactly the change starts. The type of information provided to describe a figure's overall trend varies. When the figure has multiple lines, annotators tend to specify details such as: the relative rate of change of the lines, any notable line intersections, and

[4]`https://github.com/anita76/LineCapDataset`

Table 2: Average, mode, and maximum number of words and number of references to axis values for individual line and overall chart descriptions. The minimum number of words was 3 while the minimum number of axis value references was 0 for all description types.

description type	words per annotation			values per annotation		
	mean	mode	max	mean	mode	max
line 1	15.1	6	112	0.6	0	16
line 2	14.4	6	123	0.5	0	11
line 3	13.9	6	68	0.4	0	11
line 4	13.1	6	96	0.4	0	16
line 5	12.9	6	82	0.4	0	6
overall chart	25.6	16/17	123	0.5	0	12

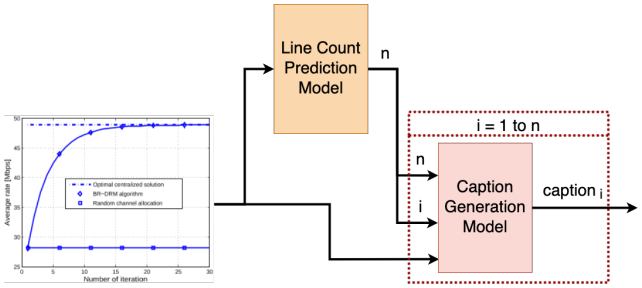


Figure 2: The deep learning pipeline for caption generation where n is the predicted number of lines in the figure, i is the line index which iterates from 1 to n, and $caption_i$ is the generated caption for line i.

the relative order of the lines along the y-axis. When the figure has a single line, the figure's overall trend either mirrors or is a summary of the description provided for the line's trend.

4 Experiments

To better understand LineCap's complexity, we set up a baseline deep learning model for predicting individual line trends. Training separate models for each of the figure lines is inefficient and does not scale well to figures with large number of lines. Furthermore, the knowledge required for generating descriptions for lines 1 through 5 is transferable between all these lines. As such, we built a model that generates descriptions for all the figure lines one at a time. We designed a two-staged deep learning pipeline that is comprised of two neural network models: a line count prediction and a caption generation model. First, the line count prediction model receives a figure as input and predicts its number of lines, n. Second, the pipeline iterates n times. At every iteration, the caption generation model produces a description for the corresponding line, which is based on the following inputs: the figure image, the iteration number (indicating the line index), and a number indicating how many lines the figure has in total.

The two models were implemented using a modified version of the PReFIL [15] model. Despite its simple architecture compared to other figure question answering and captioning models, PReFIL achieves high accuracy performance on FigureQA [16] and DVQA [14]. Furthermore, unlike models such as FigJAM [31] and STL-CQA [35], it does not require any auxiliary annotations such as the figure texts and bounding boxes – which our dataset does not supply.

4.1 Line Count Prediction

Our line count prediction model uses: a DenseNet [12] to process the figure image; two fusion blocks [15] for processing high and low level feature maps from the DenseNet; and, a neural network classifier that predicts the output.

We used 80% (2,822 figures) of our data for training, 10% (353 figures) for validation and 10% (353 figures) for testing. We trained the model using both our training data and the 40,000 line graphs

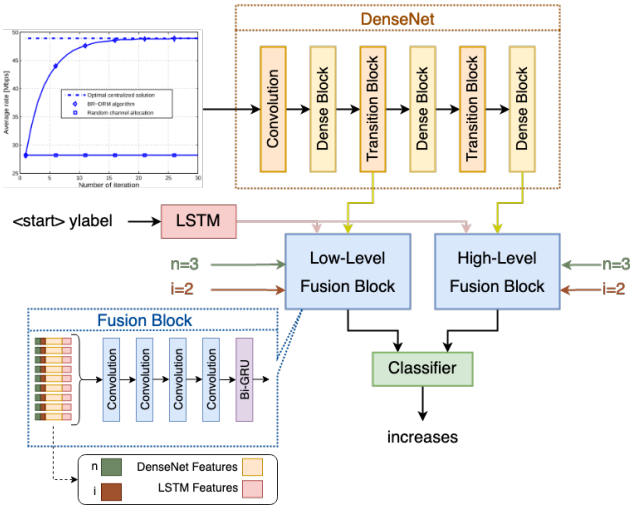


Figure 3: The architecture of the caption generation model. Several copies of the line index, i, and number of lines in the figure, n, are concatenated to the LSTM and DenseNet features, and are given as inputs to the fusion blocks. The line count prediction model has the same overall architecture but does not have an LSTM.

Table 3: Accuracy of the line count prediction model on the FigureQA and LineCap datasets when trained with the full FigureQA dataset and a sub-sample of the FigureQA dataset.

	FigureQA		LineCap	
	Validation 1	Validation 2	Validation	Test
Accuracy (%) Full training	99.88	99.88	94.90	94.05
Accuracy (%) Sub-sample training	98.50	98.39	90.93	91.50

in the training split of FigureQA. Adding FigureQA to our data, boosted the model's initial accuracy from about 70% to about 90%. Our model obtained a final accuracy of 94.05% on our test set. This is lower than the 99.88% accuracy on the line charts from FigureQA. To ensure that this discrepancy was not solely due to the much larger number of synthetic data from FigureQA, we also trained the model with our data and only a random sub-sample of 2,822 figures from FigureQA's training split. The accuracy values on the FigureQA dataset were still much greater than our dataset, thus illustrating the greater complexity of our real dataset compared to synthetic ones.

4.2 Caption Generation

Our caption generation model uses an LSTM to process the figure's caption one token at a time and predict the next token of the caption. We used beam search with $k = 32$ to generate several captions and selected the caption with the highest weighted probability as the final prediction. We incorporated the scalar inputs (i.e., line index and number of lines in the figure) into the PReFIL model by passing several copies of them into the high and low level fusion blocks.

We report the model's performance in terms of three automated machine translation metrics: BLEU-4[5] [27], CIDEr [37], and BLEURT [32]. Additionally, to better assess the accuracy of generated captions, we randomly sampled 100 figures from our test split and asked Mechanical Turk workers to rank the accuracy of the descriptions for each of the figure lines on a scale of 1 to 5, where 1 is completely inaccurate and 5 is completely accurate.

[5]We used NLTK's implementation of BLEU-4 with the smoothing 7 technique [3]

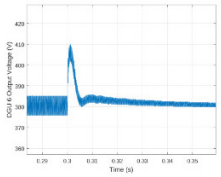
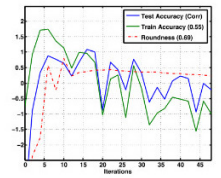
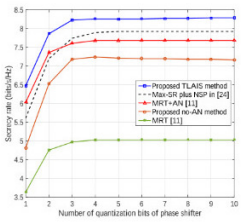
	Gold Descriptions	Generated Descriptions
	Line 1: ylabel initially expresses as a flat line. ylabel then expresses a concave down arch before increasing at a decreasing rate and flattening as xlabel increases. Line expresses as initially thick, then a thick line that increases in thickness to a medium thickness as the trend follows the concave down arch and continues as xlabel increases with a medium thickness.	Line 1: ylabel fluctuates between increasing and decreasing and appears noisy with an overall horizontal trend
	Line 1: line trends with sporadic spiking and increases and decreases overall increasing rapidly then decreasing gradually. Line 2: line trends with sporadic spiking and increases and decreases overall increasing rapidly then decreasing gradually. Line 3: line trends increasing rapidly into an upward spike then decreases and increases into another upward spike then decreases and then trends roughly horizontal.	Line 1: ylabel is at first concave up and then increases at decreasing rate and then is concave down Line 2: ylabel is at first concave up and then increases at decreasing rate and then is concave down Line 3: ylabel is at first concave up and then increases at decreasing rate and then is concave down
	Line 1: line increases at a decreasing rate and then is constant on the ylabel about _value_. Line 2: line increases at a decreasing rate and then is constant on the ylabel at _value_. Line 3: line increases at a decreasing rate and then is constant on the ylabel at _value_. Line 4: line increases at a decreasing rate and then is roughly constant on the ylabel about _value_. Line 5: line increases at a decreasing rate and then is constant on the ylabel at _value_.	Line 1: ylabel is increasing at decreasing rate until it flattens with value of around _value_ Line 2: ylabel increases at decreasing rate before flattening to _value_ as xlabel increases Line 3: ylabel is increasing at decreasing rate until it flattens with value of around _value_ Line 4: ylabel is increasing at decreasing rate until it flattens with value of around _value_ Line 5: ylabel is increasing at decreasing rate until it flattens with value of around _value_

Figure 4: Sample gold and generated line descriptions for figures with one, three, and five lines.

Table 4: The caption generation model's performance on the test split of LineCap. Accuracy is reported for a sub-sample of 100 figures. All scores are reported when the ground-truth line count number is used in the caption generation model. Scores that use the predicted line count number are in the supplementary materials.

figure with	BLEU-4	CIDEr	BLEURT-20	Accuracy
1 line	0.366	1.173	0.511	**3.43**
2 lines	0.418	1.096	0.524	3.13
3 lines	0.433	1.244	0.521	3.03
4 lines	0.443	**1.458**	0.516	3.22
5 lines	**0.455**	1.018	**0.529**	3.36
all figures	0.433	1.229	0.522	3.20

Based on human evaluations, descriptions of single-lined figures are the most accurate. The model generally does well on simple trends but struggles with more complex trends and multi-lined charts. For multi-lined charts, we observe that the model mostly repeats the same description for all of the figure lines even if the lines do not display the same trend (e.g., second example in Fig. 4). Some of the higher accuracy scores for multi-lined charts are due to all the figure lines having the same overall trend, as is the case with the last example in Fig. 4. This could be because the model is unable to learn the correlation between the line number indices and the lines in the figure. Additional figure annotations such as bounding boxes and labels could help guide the model in learning this information.

We also calculate correlation coefficients between the automated metrics and human evaluation. BLEURT has the highest correlation of 0.45. CIDEr and BLEU-4 have correlation coefficients of only 0.26 and 0.25, respectively, suggesting that these metrics are likely not suitable for effectively comparing figure captioning models.

5 Conclusion

We created LineCap, a novel dataset of line charts for figure captioning models. We subsequently established baseline line count and caption prediction performances. Through this work, we gathered the following insights and areas of future research toward automated captioning of data visualizations using deep learning models:

- Future work should aim to create datasets similar to LineCap for other chart types. Creators of datasets for automated figure captioning should focus on gathering real figures and human-written captions. Novice web-workers should be provided with further guidance on how to describe figures in order to ensure accuracy. Future research could also benefit from investigating how visually impaired audiences perceive the descriptions written by sighted individuals.

- Previous works [18] have investigated the limitations of automated metrics for evaluating *image* captioning models, and have proposed guidelines for proper assessment of these models. *Figure* captions have additional nuances that are not common in image captions. For instance, while the order in which objects in an image are described is not important, changing the order of trends in a line would lead to inaccurate descriptions. Such nuances warrant similar investigations toward identifying suitable metrics for evaluating figure captioning models. Our experiments have shown that some common automated evaluation metrics do not correlate well with human evaluation. To develop state-of-the-art figure captioning models, future research should first identify suitable automated evaluation metrics for this task.

- Deep learning models for figure captioning can benefit from incorporating additional intermediate prediction tasks from previous research on chart processing and analysis [8]. Tasks such as segmenting the lines or extracting data values from the figures could be particularly useful for distinguishing between the different lines in a multi-lined figure. Training intermediate tasks on large, synthetic datasets and fine-tuning on smaller, real datasets could significantly boost performance without the costs of creating large, crowd-sourced datasets.

Acknowledgments

We would like to thank Zhutian Chen and the anonymous reviewers for their valuable feedback. We would also like to thank Mechanical Turk workers for their contributions to the LineCap dataset. This work was supported by the Faculty Special Projects Fund from the Harvard Data Science Initiative. Some of the computations in this work were run on the FASRC Cannon cluster at Harvard University.

REFERENCES

[1] R. A. Al-Zaidy, S. R. Choudhury, and C. L. Giles. Automatic summary generation for scientific data charts. In *Proc. Workshops at AAAI Conference on Artificial Intelligence*, pp. 658–663. AI Access Foundation, Phoenix, USA, jan 2016.

[2] 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.

[3] B. Chen and C. Cherry. A systematic comparison of smoothing techniques for sentence-level BLEU. In *Proc. Workshop on Statistical Machine Translation*, pp. 362–367. Association for Computational Linguistics, Baltimore, Maryland, USA, June 2014.

[4] C. Chen, R. Zhang, E. Koh, S. Kim, S. Cohen, and R. Rossi. Figure captioning with relation maps for reasoning. In *Proc. WACV*, pp. 1537–1545. IEEE Computer Society, Los Alamitos, CA, USA, mar 2020.

[5] W. Chen, J. Chen, Y. Su, Z. Chen, and W. Y. Wang. Logical natural language generation from open-domain tables. In *Proc. Annual Meeting of the Association for Computational Linguistics*, pp. 7929–7942. Association for Computational Linguistics, Online, jul 2020.

[6] C. Clark and S. Divvala. PDFFigures 2.0: Mining figures from research papers. In *Proc. JCDL*, p. 143–152. Association for Computing Machinery, New York, USA, 2016.

[7] M. Corio and G. Lapalme. Generation of texts for information graphics. In *Proc. EWNLG*, pp. 49–58. Toulouse, France, 1999.

[8] K. Davila, S. Setlur, D. Doermann, B. U. Kota, and V. Govindaraju. Chart mining: A survey of methods for automated chart analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3799–3819, 2021.

[9] S. Demir, D. Oliver, E. Schwartz, S. Elzer, S. Carberry, and K. F. McCoy. Interactive SIGHT demo: Textual summaries of simple bar charts. In *Proc. International ACM SIGACCESS Conference on Computers and Accessibility*, p. 267–268. Association for Computing Machinery, New York, USA, 2010.

[10] D. Gurari, Y. Zhao, M. Zhang, and N. Bhattacharya. Captioning images taken by people who are blind. In A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds., *Proc. Computer Vision – ECCV*, pp. 417–434. Springer International Publishing, Cham, Switzerland, 2020.

[11] T.-Y. Hsu, C. L. Giles, and T.-H. Huang. SciCap: Generating captions for scientific figures. In *Proc. EMNLP*, pp. 3258–3264. Association for Computational Linguistics, Punta Cana, Dominican Republic, Nov. 2021.

[12] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *Proc. CVPR*, pp. 2261–2269. IEEE Computer Society, 2017.

[13] H. Jhamtani and T. Berg-Kirkpatrick. Truth-conditional captions for time series data. In *Proc. EMNLP*, pp. 719–733. Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021.

[14] K. Kafle, B. Price, S. Cohen, and C. Kanan. DVQA: Understanding data visualizations via question answering. In *Proc. CCVPR*, pp. 5648–5656. IEEE Computer Society, Salt Lake City, USA, 2018.

[15] K. Kafle, R. Shrestha, S. Cohen, B. Price, and C. Kanan. Answering questions about data visualizations using efficient bimodal fusion. In *Proc. WACV*, pp. 1498–1507. IEEE Computer Society, 2020.

[16] S. E. Kahou, V. Michalski, A. Atkinson, Ákos Kádár, A. Trischler, and Y. Bengio. FigureQA: An annotated figure dataset for visual reasoning. In *Proc. ICLR*. OpenReview, Vancouver, Canada, 2018.

[17] S. Kantharaj, R. T. K. Leong, X. Lin, A. Masry, M. Thakkar, E. Hoque, and S. Joty. Chart-to-Text: A large-scale benchmark for chart summarization. In *Proc. Annual Meeting of the Association for Computational Linguistics*, 2022.

[18] M. Kilickaya, A. Erdem, N. Ikizler-Cinbis, and E. Erdem. Re-evaluating automatic metrics for image captioning. In *Proc. Conference of the European Chapter of the Association for Computational Linguistics*, pp. 199–209. Association for Computational Linguistics, Valencia, Spain, Apr. 2017.

[19] D. H. Kim, V. Setlur, and M. Agrawala. Towards understanding how readers integrate charts and captions: A case study with line charts. In *Proc. CHI Conference on Human Factors in Computing Systems*, CHI '21. Association for Computing Machinery, New York, NY, USA, 2021.

[20] S. Li, Z. Tao, K. Li, and Y. Fu. Visual to text: Survey of image and video captioning. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 3(4):297–312, 2019.

[21] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds., *Proc. Computer Vision – ECCV*, pp. 740–755. Springer International Publishing, Cham, Switzerland, 2014.

[22] A. Lundgard and A. Satyanarayan. Accessible visualization via natural language descriptions: A four-level model of semantic content. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):1073–1083, 2022.

[23] V. O. Mittal, J. D. Moore, G. Carenini, and S. Roth. Describing complex charts in natural language: A caption generation system. *Computational Linguistics*, 24(3):431–467, 1998.

[24] P. Moraes, G. Sina, K. McCoy, and S. Carberry. Generating summaries of line graphs. In *Proc. INLG*, pp. 95–98. Association for Computational Linguistics, Philadelphia, USA, June 2014.

[25] V. S. Morash, Y.-T. Siu, J. A. Miele, L. Hasty, and S. Landau. Guiding novice web workers in making image descriptions using templates. *ACM Trans. Access. Comput.*, 7(4), nov 2015.

[26] J. Obeid and E. Hoque. Chart-to-Text: Generating natural language descriptions for charts by adapting the transformer model. In *Proc. INLG*, pp. 138–147. Association for Computational Linguistics, Dublin, Ireland, Dec. 2020.

[27] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proc. Annual Meeting of ACL*, pp. 311–318. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, July 2002.

[28] A. Parikh, X. Wang, S. Gehrmann, M. Faruqui, B. Dhingra, D. Yang, and D. Das. ToTTo: A controlled table-to-text generation dataset. In *Proc. EMNLP*, pp. 1173–1186. Association for Computational Linguistics, Online, Nov. 2020.

[29] B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proc. ICCV*, pp. 2641–2649. IEEE Computer Society, NW Washington, USA, 2015.

[30] J. Poco and J. Heer. Reverse-engineering visualizations: Recovering visual encodings from chart images. *Comput. Graph. Forum*, 36(3):353–363, jun 2017.

[31] X. Qian, E. Koh, F. Du, S. Kim, J. Chan, R. A. Rossi, S. Malik, and T. Y. Lee. Generating accurate caption units for figure captioning. In *Proc. International World Wide Web Conference*, pp. 2792–2804. Association for Computing Machinery, New York, USA, 2021.

[32] T. Sellam, D. Das, and A. Parikh. BLEURT: Learning robust metrics for text generation. In *Proc. Annual Meeting of ACL*, pp. 7881–7892. Association for Computational Linguistics, Online, July 2020.

[33] K. Seweryn, K. Lorenc, A. Wróblewska, and S. Sysko-Romańczuk. What will you tell me about the chart? – automated description of charts. In T. Mantoro, M. Lee, M. A. Ayu, K. W. Wong, and A. N. Hidayanto, eds., *Proc. Neural Information Processing*, pp. 12–19. Springer International Publishing, Cham, Switzerland, 2021.

[34] N. Siegel, Z. Horvitz, R. Levin, S. Divvala, and A. Farhadi. FigureSeer: Parsing result-figures in research papers. In B. Leibe, J. Matas, N. Sebe, and M. Welling, eds., *Proc. Computer Vision – ECCV*, pp. 664–680. Springer International Publishing, Cham, Switzerland, 2016.

[35] H. Singh and S. Shekhar. STL-CQA: Structure-based transformers with localization and encoding for chart question answering. In *Proc. EMNLP*, pp. 3275–3284. Association for Computational Linguistics, Online, Nov. 2020.

[36] A. Spreafico and G. Carenini. Neural data-driven captioning of time-series line charts. In *Proc. AVI*, pp. 1–5. Association for Computing Machinery, New York, USA, 2020.

[37] R. Vedantam, C. L. Zitnick, and D. Parikh. CIDEr: Consensus-based image description evaluation. In *Proc. CVPR*, pp. 4566–4575. IEEE Computer Society, 2015.