# Understanding the role of structured information for answering questions on data visualizations

Ankur Garg* Uday Kusupati* Venkata Ravi Teja Ailavarapu*
Dept. of Computer Science
The University of Texas at Austin
{ankgarg,uday,avrteja}@cs.utexas.edu

## Abstract

*Documents contain data visualizations to summarize large pieces of information effectively. In the pursuit of true document understanding, it is important to understand these visualizations in documents as they can help in tasks like summarization and question answering. These visualizations have an inherent structure and this paper attempts to leverage this structural information in the visualizations to improve question answering performance.*

*We specifically focus on bar charts and use bounding box information given in the dataset to identify the graphic and text elements separately. This allows us to learn separate features for the two different types of elements which is in stark contrast to the current practice of using a single feature for the whole visualization using models pre-trained on ImageNet dataset. Moreover, we use the multi-head attention mechanism to attend to these different features in the context of the given question. Our experiments on a subset of the DVQA dataset show that using structural information gives significantly better results than the current baseline models and using multi-head attention mechanism improves the capability of the model to answer complex questions.*

## 1. Introduction

Document understanding has always been an active area of research. This has given rise to problems like summarization [3], information extraction [6], and machine reading comprehension [4, 20]. Scientific documents from repositories like PubMed[1] form an important corpus for many of these problems. In such documents, data visualizations, e.g. bar charts, pie charts, and plots, play an important role in summarizing large amounts of information effectively. Readers use these visualizations to get a quick understanding of the information in the scientific documents, e.g. results of an experiment, distributional characteristics of the dataset etc. Thus, in order to build a holistic understanding of the document, it is vital to understand these data visualizations.

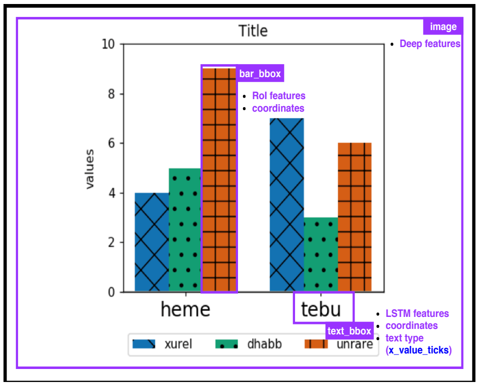


Figure 1: Structural information present in a bar chart

Recent advances in Visual Question Answering (VQA) [15, 16] have sparked an interest in understanding these visualizations. VQA systems can answer complex natural language questions based on a given image. The release of large scale datasets like FigureQA [12] and Data Visualization Question Answering (DVQA) [10] has made it possible to leverage deep learning based models, originally proposed for VQA, for understanding data visualizations and answer questions on them. However, the baselines presented for these datasets [10, 12] use models [18, 23] that consider the visualization as a single image. There is no attempt to make use of the structure present in these visualizations and treat the text elements in the visualization differently from the graphic elements.

In this paper, we aim to *leverage the structured information present in data visualizations to build an effective question answering system for data visualizations.* For this paper, we only consider bar chart visualizations. Figure 1 shows the structural information present in a bar chart. Bars

*All the authors contributed equally to the project

[1] https://www.ncbi.nlm.nih.gov/pubmed/

are graphic elements and the bounding box information can help identify each bar individually as well as the value depicted by the bar. Similarly, the text elements in the bar chart can also be identified and the text can be extracted using standard OCR packages. In comparison, prior approaches only consider the visualization as a single image and use CNN features pre-trained on ImageNet [5] dataset. However, we use individual features for all different elements in the visualization in addition to the CNN features for the whole image. We use image features for each of the graphic elements in the bar chart and similarly, use text features for the text elements. We pass all these features to a model that uses a stacked attention mechanism to attend to these features in the context of the given question to find the correct answer. We argue that using such rich structural information improves the model's ability to establish relation between different elements in the bar chart which is specifically crucial for answering complex reasoning-based questions. Moreover, we propose a multi-head attention mechanism [22] to attend to these features that helps in capturing multiple aspects of the same feature and further improve the model's understanding of the visualization.

We make the following contributions in this paper:

1. We propose an approach that exploits information about the structure of the data visualizations. Our model treats the graphical elements in the visualization (like bars in bar charts) differently than the text elements (legend and titles). Different features are extracted for different types of elements instead of passing a single feature representation for the whole visualization.

2. We use the attention mechanism to attend to different features of the visualization in the context of the given question. Our approach is flexible to accommodate different attention mechanisms and we find, that a multi-head attention mechanism performs better than a single attention mechanism.

3. We test the efficacy of our approach on the DVQA dataset [10] and our experiments show that using information about the structure of the visualizations actually helps in improving the model's understanding and ultimately its performance on the question answering task. We show that our approach outperforms the best performing baseline proposed by the DVQA dataset authors by a significant margin.

The rest of the paper is organized as follows: Section 2 describes the prior art and Section 3 explains the solution architecture. Section 4 presents the experimental setup and the results. Section 5 concludes the paper.

## 2. Related Work

In this section, we provide an overview of related work in the area of VQA and figure understanding.

### 2.1. Visual Question Answering (VQA)

There is an active interest in the community for developing VQA systems to answer natural language questions about an image. This interest has led to emergence of several datasets [2, 17] and sophisticated models ranging from Bayesian approaches [11] to neural attention based methods [23, 14]. There are certain differences between understanding images and visualizations that limit the direct application of these models in our setting. First, all these models consider the visualization image as one single entity and use CNN features pre-trained on ImageNet [5] dataset. It is a potential problem since the answer to many questions on a visualization can come directly from the text in them and this information is lost when we treat it as a single image. This makes it extremely difficult for the model to recover the actual text through these pre-trained features. Second, the problem of out-of-vocabulary (OOV) words is more pronounced in case of visualizations than images in the VQA datasets [2, 17]. Since different visualizations capture different information, thus, the same text labels are not encountered in many visualizations in a realistic setting. Thus, effective question answering systems for visualizations need to be more robust in handling OOV words both in the question as well as the answer.

### 2.2. Figure Understanding

We are trying to address a problem that falls under the area of figure understanding. Figure understanding refers to being able to answer reasoning based questions for a given figure. The answers to these questions require more complex inference skills than simply looking at the graphic elements in the figure. The CLEVR [9] dataset is one of the more popular datasets for such problems. However, the lack of multiple-choice questions in our setting makes the task harder as now the models need to come up with open ended answers. There also has been some prior work specifically for extracting information from bar charts [1, 19]. But these works only focus on extracting the information from the visualization and do not address the problem of answering reasoning-based questions on them.

## 3. Solution Architecture

Figure 5a shows our solution architecture. We take motivation from the stacked attention network (SAN) [23] architecture for VQA. The SAN model consists of an image model, a question model, and a stacked attention network. The image and question models produce feature representations of the input image and question. The first attention

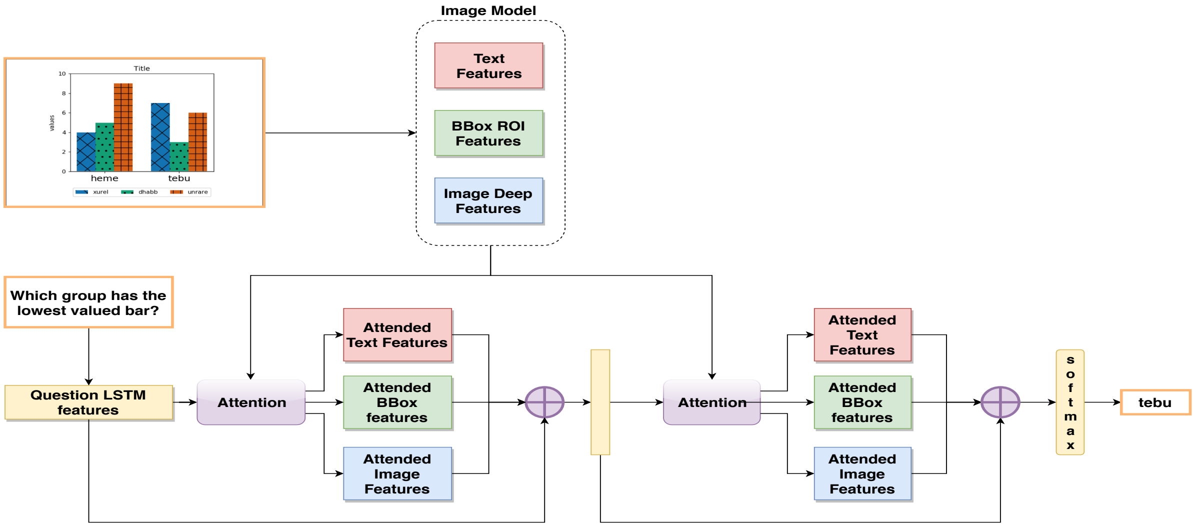


Figure 2: Solution Architecture

module attends over the different image features using the question and produces a more informative representation. The second (stacked) attention module attends over the image features using this new representation. The output of the final layer is passed through a linear softmax layer to get the final answer. In the following subsections, we explain the individual feature extraction modules and the two types of attention mechanisms.

## 3.1. Image Model

The image model in the SAN network passes the image through a CNN (like VGG [21] or ResNet [8]). We pass the 448 × 448 × 3 image through a 16 layer VGG network pretrained on the ImageNet dataset. We use the image features ($f_I$) from the last pooling layer of the final convolutional block. The size of $f_I$ will be 196 × 512 which can be thought of as features in dimension $\mathbb{R}^{512}$ for regions in a 14 × 14 image.

$$f_I = \text{VGG16}_{\text{pool5}}(I)$$

We term these feature representations as global image features. In addition to these, we compute three other features that capture structural information in the images.

### 3.1.1 Bounding Box RoI Features

The main structure present in the data is the bounding box information for the bars in graphs (refer Figure1). We use these bounding boxes as regions of interest and perform region of interest pooling (RoI pooling) [7] over the image features $f_I$. The maximum number of bars over all images in the dataset is 30. Therefore, the output of this layer will be of size 30 × 512 × 7 × 7 (masked appropriately). To reduce the dimension of the output we apply an average pooling layer on top of this to obtain a final feature vector of size 30 × 512.

$$f_{RoI} = \text{ROI\_POOL}(f_I, bboxes)$$

### 3.1.2 Text Features

An image consists of multiple textual annotations (elements). There are six types of text elements present in a bar chart: $\{$*legend_heading, legend, value_ticks, title, x_ticks, y_ticks*$\}$. We encode the texts in the image using the dynamic encoding model (DEM) [10]. DEM maps each text in the image to a unique numeric index. DEM computes these mappings using a deterministic algorithm over the bounding boxes of the texts. This helps in dynamically adding the words from the image by aliasing them with indices based on their position in the image instead of the actual text itself.

We represent these DEM encoded texts using indicator features ($I_{text}$) that are passed to an embedding layer of embedding size $t$. Further, we also compute an indicator feature for the text type ($I_{type}$) and concatenate it with the DEM text embedding.

The maximum number of texts in an image are 30. Therefore, the output of the embedding layer will be 30 × $t$ (masked appropriately). The final representation of the texts in the image after concatenating the type information will be of size 30 × $(t + 6)$.

$$f_{text} = [W_t \cdot I_{text}; I_{type}]$$

### 3.1.3 Position Features

The features discussed in sections 3.1.1 and 3.1.2 do not contain any information about the relative or absolute positions of the bars or texts. Therefore, we also concatenate the bounding box RoI and text representations with absolute bounding box positions. In the following equations, $x, y$ denote the coordinates of the top-left corner while $w, h$ denote the width and height of the bounding box.

$$
\begin{aligned}
f_{text+pos} &= [f_{text}; x; y; w; h] \\
f_{RoI+pos} &= [f_{RoI}; x; y; w; h]
\end{aligned}
$$

## 3.2. Question Model

Given a set of questions from the training set, we generate a vocabulary consisting of words from the question. We remove those words from the vocabulary, which are also present as text on a bar chart (chart-specific words). We represent these chart-specific words by the bounding box indices returned by DEM as explained in section 3.1.2.

Given a question $Q = [q_1, q_2, .., q_n]$ denoted by indicator features, we compute an embedding of size $q$ using an embedding matrix $W_q$. We pass these embedded vectors through a LSTM and represent the question using the output of the last timestep of the final layer of the LSTM.

$$
\begin{aligned}
x_i &= W_q \cdot q_i \text{ for } i \in 1, 2, .., n \\
f_Q &= \text{LSTM}([x_1, x_2, .., x_n])
\end{aligned}
$$

## 3.3. Attention Mechanism

We describe two different attention mechanisms (modules) in the following subsections. Each attention module is used in a stacked fashion to fetch information from the feature set for a given question and image.

### 3.3.1 Simple Attention

Given the image features, described in Section 3.1, along with the question embedding, we query the relevant features to the question by attending multiple times over the features. We project the image features $f_I$, RoI features $f_{RoI}$ (or $f_{RoI+pos}$), and the text features $f_{text}$ (or $f_{text+pos}$) to the space of the question embedding $f_Q$ using a linear layer. Then, we compute attention weights for the bounding box features as follows.

$$
\begin{aligned}
a_{RoI} &= \tanh(W_{RoI} \cdot f^p_{RoI} + (W_Q \cdot f_Q + b)) \\
p_{RoI} &= \text{softmax}(W_P \cdot a_{RoI} + b_P) \\
\tilde{f}_{RoI} &= \sum_i p_{RoI_i} \cdot f^p_{RoI_i} \\
u &= f_Q + \tilde{f}_I + \tilde{f}_{RoI} + \tilde{f}_{text}
\end{aligned}
$$

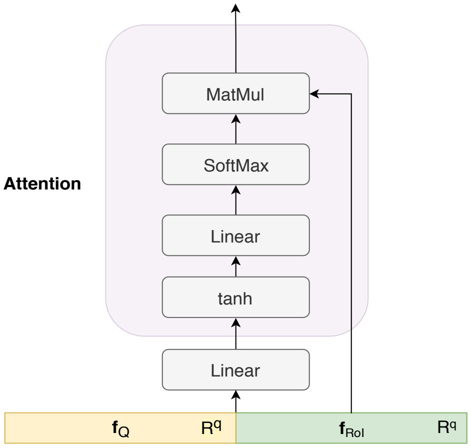


Figure 3: Illustration of simple attention mechanism. $f_{RoI}$ in the image represents the projected features. The output is $\tilde{f}_{RoI}$.

This is illustrated in Figure 3. Here $f^p_{RoI}$ are the projected features and $f^p_{RoI_i}$ corresponds to the features of one bounding box. Each weight $p_{A_i}$ corresponds to the attention weight of $f^p_{RoI_i}$ and we add the weighted features back to the question embedding. We compute attention weights for the other features in a similar fashion and add the weighted features back to the question embedding. Given this new embedding, we iterate this process multiple times, thus stacking layers of attention on top of each other. Formally,

$$
\begin{aligned}
a^k_{RoI} &= \tanh(W^k_{RoI} \cdot f^p_{RoI} + (W^k_Q \cdot u^{k-1} + b^k)) \\
p^k_{RoI} &= \text{softmax}(W^k_P \cdot a^k_{RoI} + b^k_P) \\
\tilde{f}^k_{RoI} &= \sum_i p^k_{RoI_i} \cdot f^p_{RoI_i} \\
u^k &= u^{k-1} + \tilde{f}^k_I + \tilde{f}^k_{RoI} + \tilde{f}^k_{text}
\end{aligned}
$$

We do this for $K$ times ($K = 2$ in our case) and obtain the answer from a softmax layer.

$$
p_{ans} = \text{softmax}(W_u \cdot u^K + b_u)
$$

### 3.3.2 Multi-head Attention

We use the idea of using multiple heads from literature [22] to jointly attend to different kinds of features that might be complementary and get lost when a single head is used due to averaging. We use multi-head attention over different types of features to obtain the attended features and add them back to the question representation just like before. The attention for RoI features is illustrated in Figure 4. For-

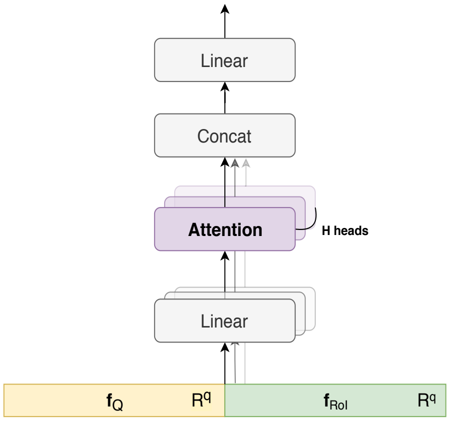


Figure 4: Multi-head attention. $f_{RoI}$ in the image represents the projected features. The output is $\tilde{f}_{RoI}$.

mally, at stack level $k$ and attention head $h$,

$$\begin{aligned} {}^{h}a^{k}_{RoI} &= \tanh({}^{h}W^{k}_{RoI} \cdot f^{p}_{RoI} \oplus ({}^{h}W^{k}_{Q} \cdot u^{k-1} + {}^{h}b^{k})) \\ {}^{h}p^{k}_{RoI} &= \text{softmax}({}^{h}W^{k}_{P} \cdot a^{k}_{RoI} + {}^{h}b^{k}_{P}) \\ \tilde{f}^{k}_{RoI} &= \tanh(W^{o}_{RoI} \cdot [{}^{1}p^{k}_{RoI_i} \cdot f^{p}_{RoI_i}; ...; {}^{H}p^{k}_{RoI_i} \cdot f^{p}_{RoI_i}]) \\ u^{k} &= u^{k-1} + \tilde{f}^{k}_{I} + \tilde{f}^{k}_{RoI} + \tilde{f}^{k}_{text} \end{aligned}$$

Here $H$ ($= 3$ in our case) is the total number of heads. The final output $u^K$ is passed through a linear layer with softmax activation as before to obtain the prediction.

# 4. Experimental Evaluation

## 4.1. Dataset

We perform our experiments on the DVQA dataset [10]. It is a synthetically generated dataset consisting of more than 3 million question-answer pairs for $300,000$ bar chart images. The questions can belong to one of the three categories: structure understanding, data retrieval or reasoning. The structure understanding category consists of questions on the overall structure of the bar chart while the data retrieval category contains questions pertaining to fetching information from specific parts of the bar chart. Questions in the reasoning category require focusing on multiple parts of the bar chart and drawing inferences from them. Reasoning based questions are arguably the most difficult category of questions among the three categories mentioned above.

Further, three splits of the dataset are provided: *train*, *val easy* and *val hard*. The difference between the *val easy* and *val hard* set is the vocabulary that generates the text labels in the bar charts. For all the bar charts in the sets *train* and *val easy*, the DVQA [10] authors use the same vocabulary to generate the text labels. However, the text labels in the *val hard* set are completely different and thus, unseen during training. Due to the sheer size of the dataset, we use a $1\%$ random subset of the all the three splits. All future references of the dataset in the paper refer to these subsets of dataset for the three splits. The basic distributional characteristics of the subsets of each of the splits of the datasets is given in Tables 1 and 2. It is important to note that the distributional characteristics of the subset resemble closely with that of the original dataset.

| | Images | Questions |
|---|---|---|
| *train* | 1992 | 23,253 |
| *val easy* | 500 | 5,805 |
| *val hard* | 497 | 5,813 |
| **Total** | **2,989** | **34,871** |

Table 1: DVQA dataset statistics for the three splits

| | Structure | Data | Reasoning |
|---|---|---|---|
| *train* | 3,148 | 7,519 | 12,586 |
| *val easy* | 751 | 1,878 | 3,176 |
| *val hard* | 805 | 1,829 | 3,179 |
| **Total** | **4,704** | **11,226** | **18,941** |

Table 2: Question Type Distribution across the three splits of the dataset

## 4.2. Experiment Details

We train all the models with the same hyper-parameters to make a fair comparison. We use the Adam [13] optimizer with a learning rate of $0.001$. After the first 30 epochs, we decrease the learning rate by half every 10 epochs. We use $20\%$ of the *train* set as a holdout set for evaluating the training across different epochs. We evaluate the model based on the accuracy of the answer for a given question and bar chart image. Thus, we train all models in our experiments using the early stopping criterion by observing the accuracy on the holdout set.

We train all our models on one Nvidia® GTX 1080 GPU and the average time to complete one epoch during training is about 15 minutes. The code is available at: `https://github.com/ankurgarg101/plot_qa/`.

## 4.3. Baselines

Since, DVQA [10] is a relatively new dataset, the best performing baseline is the Stacked Attention Network with DYnamic Encoding (SANDY) model proposed by the DVQA authors itself. This model is an adaptation of the originally proposed version of SAN [23] for VQA.

## 4.4. Results

We discuss the results of our experiments on the DVQA dataset in this section. We perform an ablation study to understand the importance of structural features as well as compare the two attention mechanisms described in Section 3.3. The results of the experiments on the *val easy* and *val hard* sets are given by Tables 3 and 4 respectively.

The first takeaway from Tables 3 and 4 is that all variants of our proposed approach perform significantly better than the SANDY baseline. This justifies our claim in Section 1 that using structural information gives a richer form of supervision to model to aid its understanding of the bar chart. If we breakdown the accuracy achieved by question type, we see that the our model architecture improves the most on data retrieval questions. This is due to the fact that the answers to many of the data retrieval questions come from the text elements in the bar charts. Obtaining text features for those elements makes it easier for the model to establish the relation between the question and text in the bar chart. Moreover, similar accuracies on *val easy* and *val hard* sets justifies that our model is quite robust to out-of-vocabulary problem.

We perform an ablation study by varying the features used to represent the bar chart image to identify the most important features. Needless to say, using the text feature alone on top of the whole image features gives the most improvement in accuracy. This is due to the same reason mentioned before that many answers directly come from the text in these bar charts. Interestingly, as we add the RoI and Positional features, the accuracy for reasoning-based questions shows a gradual improvement which is encouraging. We believe that this is due to the fact that reasoning-based questions require establishing relations between different elements in the bar chart (both graphical and text). Hence, providing separate features for all the elements helps the model as it doesn't need to recover them from the features of the whole image. Finally, we see that the best performing models use features for all different elements as well as the whole image. A possible reason behind this observation can be the fact that the features for the whole image provide a good context for all elements and aid the model in establishing relations between them.

Next, we compare the different attention mechanisms proposed in Section 3.3. We use three heads in the multi-head attention mechanism for all model ablations. Observing from Tables 3 and 4, we see that models using multi-head attention perform better than the ones using simple attention for the same set of features. This is in line with the improvement observed in literature by using multi-head attention [22]. This shows that the relations between elements in bar-graph images are complex enough to require multiple heads of attention for learning them. Surprisingly, most of the improvement is a result of the improvement in answering data retrieval questions. We leave further qualitative analysis of these improvements as a future work.

## 5. Conclusion and Future Work

In this paper, we present a method to leverage structural information present in bar charts for better understanding of data visualizations. We verify our claims empirically on a subset of the DVQA dataset and show significant improvements on the question answering task. Our models are particularly better at answering data retrieval questions compared to the baseline SANDY model. We also show that multi-head attention performs better than a simple single head attention mechanism.

Currently, our proposed approach uses the ground truth labels of the bounding boxes. In a real scenario, these might not be available and region proposals will be needed in this scenario. Experimenting the proposed model architecture on the FigureQA dataset will prove its robustness and we leave this to future work.

## 6. Acknowledgements

The authors would like to thank Prof. Philipp Krähenbühl for providing constructive feedback and guidance during the course of the work.

## References

[1] R. A. Al-Zaidy and C. L. Giles. Automatic extraction of data from bar charts. In *Proceedings of the 8th International Conference on Knowledge Capture*, page 30. ACM, 2015.

[2] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015.

[3] J. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336. ACM, 1998.

[4] D. Chen, A. Fisch, J. Weston, and A. Bordes. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*, 2017.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. Ieee, 2009.

[6] J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 363–370, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[7] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE con-*

| | **Features** | | | | **Accuracy (%)** | | | |
|---|---|---|---|---|---|---|---|---|
| | **Image** | **Text** | **RoI** | **Positional** | **Structure** | **Data** | **Reasoning** | **Overall** |
| SANDY | ✓ | ✗ | ✗ | ✗ | 84.42 | 31.68 | 28.74 | 36.90 |
| Simple Attention | ✓ | ✓ | ✗ | ✗ | 86.41 | 53.03 | 31.55 | 45.60 |
| | ✗ | ✓ | ✓ | ✗ | 85.49 | 47.55 | 32.75 | 44.36 |
| | ✗ | ✓ | ✓ | ✓ | 82.29 | 47.23 | 33.69 | 44.34 |
| | ✓ | ✓ | ✓ | ✓ | 86.28 | 52.44 | 34.66 | 47.10 |
| Multi-head Attention | ✗ | ✓ | ✓ | ✓ | 81.22 | 58.41 | 34.70 | 48.39 |
| | ✓ | ✓ | ✓ | ✓ | **81.76** | **63.47** | **35.29** | **50.42** |

Table 3: Performance of different model ablations on the *val easy* dataset. (✓) denotes the presence of the feature in the model while (✗) denotes the absence of it.

| | **Features** | | | | **Accuracy (%)** | | | |
|---|---|---|---|---|---|---|---|---|
| | **Image** | **Text** | **RoI** | **Positional** | **Structure** | **Data** | **Reasoning** | **Overall** |
| SANDY | ✓ | ✗ | ✗ | ✗ | 80.74 | 31.71 | 27.65 | 36.59 |
| Simple Attention | ✓ | ✓ | ✗ | ✗ | 85.09 | 54.45 | 30.67 | 45.69 |
| | ✗ | ✓ | ✓ | ✗ | 84.84 | 49.59 | 32.59 | 45.17 |
| | ✗ | ✓ | ✓ | ✓ | 82.98 | 47.56 | 34.51 | 45.33 |
| | ✓ | ✓ | ✓ | ✓ | 83.33 | 50.47 | 35.26 | 46.71 |
| Multi-head Attention | ✗ | ✓ | ✓ | ✓ | 83.11 | 57.57 | 34.57 | 48.53 |
| | ✓ | ✓ | ✓ | ✓ | **82.98** | **61.51** | **35.80** | **50.42** |

Table 4: Performance of different model ablations on the *val hard* dataset. (✓) denotes the presence of the feature in the model while (✗) denotes the absence of it.

*ference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick, and R. Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1988–1997. IEEE, 2017.

[10] K. Kafle, S. Cohen, B. Price, and C. Kanan. Dvqa: Understanding data visualizations via question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5648–5656, 2018.

[11] K. Kafle and C. Kanan. Answer-type prediction for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4976–4984, 2016.

[12] S. E. Kahou, V. Michalski, A. Atkinson, A. Kadar, A. Trischler, and Y. Bengio. Figureqa: An annotated figure dataset for visual reasoning. *arXiv preprint arXiv:1710.07300*, 2017.

[13] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[14] J. Lu, J. Yang, D. Batra, and D. Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297, 2016.

[15] M. Ren, R. Kiros, and R. Zemel. Exploring models and data for image question answering. In *Advances in neural information processing systems*, pages 2953–2961, 2015.

[16] M. Ren, R. Kiros, and R. Zemel. Exploring models and data for image question answering. In *Advances in neural information processing systems*, pages 2953–2961, 2015.

[17] M. Ren, R. Kiros, and R. Zemel. Exploring models and data for image question answering. In *Advances in neural information processing systems*, pages 2953–2961, 2015.

[18] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. In *Advances in neural information processing systems*, pages 4967–4976, 2017.

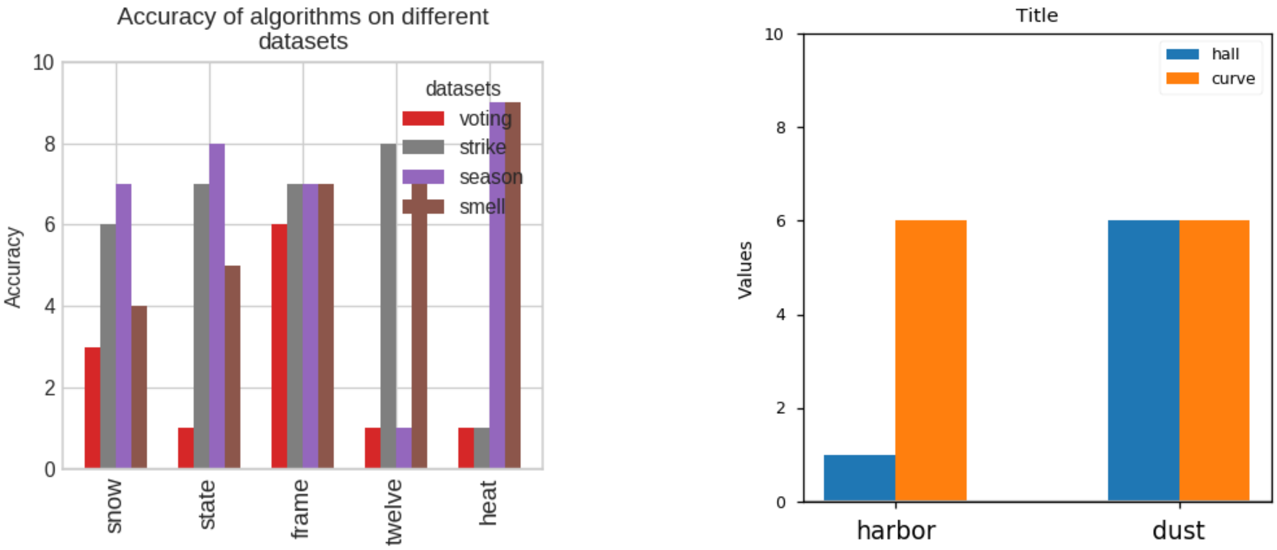
[19] M. Savva, N. Kong, A. Chhajta, L. Fei-Fei, M. Agrawala, and J. Heer. Revision: Automated classification, analysis and redesign of chart images. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 393–402. ACM, 2011.

[20] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*, 2016.

[21] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[23] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola. Stacked attention networks for image question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–29, 2016.



(a) **Q**: What is the label of the fifth group of bars from the left?
**SANDY**: season ✗
**Multi-Head-Attn**: heat ✓
**Q**: Is each bar a single solid color without patterns?
**SANDY**: yes ✓
**Multi-Head-Attn**: yes ✓

(b) **Q**:What element does the darkorange color represent?
**SANDY**: 8 ✗
**Multi-Head-Attn**: curve ✓
**Q**: Which group of bars contains the smallest valued individual bar in the whole chart?
**SANDY**: harbor ✓
**Multi-Head-Attn**: harbor ✓

Figure 5: Example results on DVQA dataset for our model using multi-head attention in comparison to SANDY