ReasonVQA: A Multi-hop Reasoning Benchmark with Structural Knowledge for Visual Question Answering – Supplementary material

Duong T. Tran[1,2]
Duong.Tran@de.bosch.com

Trung-Kien Tran[1]
TrungKien.Tran@de.bosch.com

Manfred Hauswirth[2,3]
manfred.hauswirth@tu-berlin.de

Danh Le Phuoc[2,3]
danh.lephuoc@tu-berlin.de

[1]Bosch Center for AI, Germany [2]Technical University of Berlin [3]Fraunhofer FOKUS

1. The ReasonVQA Framework Additional Details

This is additional content for Section 3 in the main paper. Here, we provide more details regarding the question generation process, the integration of external knowledge, and the visualization of answer distribution balancing and dataset splitting.

1.1. Template-based Question Generation

Our framework consists of three steps: (1) External Knowledge Integration, (2) Question Generation, and (3) Dataset Construction. In addition to Figure 2 in the main paper, Algorithm 1 also describes the detailed workflow of these steps.

Algorithm 1: Algorithm for generating questions and answers from annotated images

Input : Annotated image **Img**
Output: A set of questions **Q** generated for **Img**

/* Step 1: External Knowledge Integration (Main Paper Section 3.1) */
$\{\mathcal{C}_i\}$ = set of Wikidata entities corresponding to annotated objects in **Img**
$\mathcal{G}_i(\mathcal{V}_i, \mathcal{E}_i)$ = knowledge graph from Wikidata with root $\mathcal{C}_i$
$\mathcal{E} = \{\mathcal{E}_i\}$ // set of potential properties

/* Step 2: Question Generation (Main Paper Section 3.2) */
$\mathcal{T}_m$ = set of main templates $\forall e_i \in \mathcal{E}$
$\mathcal{T}_s$ = set of sub-clause templates $\forall e_i \in \mathcal{E}$
Function `Generate`(e_j):
    if $j = 0$ **then**
        $t \leftarrow \mathcal{T}_m[e_j]$
    else
        $t \leftarrow \mathcal{T}_s[e_j]$
    return $t \cup$ `Generate`(e_{j+1})

$\mathcal{D}_1$ = empty dataset
foreach $v \in \mathcal{V}_i$ **do**
    $\{e_j\}$ = set of edges from v to $\mathcal{C}_i$
    $\mathcal{D}_1 = \mathcal{D}_1 \cup$ `Generate`(e_0)

/* Step 3: Dataset Construction (Main Paper Section 3.3) */
$\mathcal{D}_2 = balance(\mathcal{D}_1)$ // balance the answer distribution
$\mathcal{D}_3 = split(\mathcal{D}_2)$ // split into train set and test set

1.2. Concept Linking and Template Construction

Concept Linking between Image and Knowledge Base. The process of linking an annotated object in an image to a concept in a knowledge base may vary depending on the computer vision (CV) dataset and the knowledge base (KB) used.

For Visual Genome (VG) [6], we leverage the WordNet [11] synset names provided in the annotations to identify the corresponding entity in Wikidata. Specifically, for each object associated with a synset name, we convert the synset name into a synset ID using NLTK. We then query the respective Wikidata entity via SPARQL. Figure 1 shows an example of linking the object *traffic light* in the image to the corresponding concept with the same name in Wikidata. Since the bounding box of the traffic light was annotated with the synset name `\traffic_light.n.01"`, we convert it into synset ID `\06887235-n"` using the NLTK [2] package and then search for the Wikidata entity associated with this synset ID via SPARQL.

For Google Landmarks Dataset v2 (GLDv2) [14], from the Wikimedia URLs provided in the annotations, we

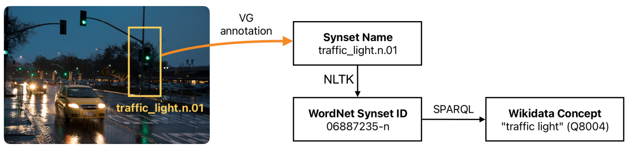

Figure 1. Example of linking an object from VG to a concept in Wikidata using Wordnet synset name. The Wikidata entity is retrieved by the WordNet synset ID, which is converted from the synset name using the NLTK package.

heuristically extract the name of the landmark. Then we search for the Wikidata concept by this name. In Figure 2, we extract the name *Maria Magdalena kyrka, Stockholm* from the Wikimedia URL. With a simple SPARQL query, we can search for the entity that links to the Wikimedia Commons resource with this name.

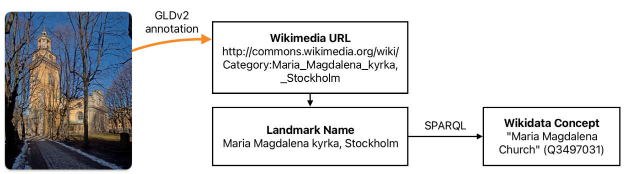

Figure 2. Example of linking a landmark from GLDv2 to a concept in Wikidata. The Wikidata entity is retrieved by its name, which is extracted from the Wikimedia URL in GLDv2.

Main Template and Sub-clause Template Crafting. After connecting an object in the image to an entity in the KB, referred to as the *root concept*, we begin gathering multi-hop knowledge. Initially, we retrieve knowledge around the root concept in the form of triplets. Each triplet corresponds to a property and connects the root concept to either a literal value or another concept. If the end of a triplet is another concept, we continue gathering knowledge for this one. This process of traversing through the knowledge graph yields multi-hop knowledge. In practice, we find that traversing up to three hops strikes a balance between complexity and minimizing grammatical errors in generated questions.

During the process of fetching knowledge from the KB, we also collect all potential properties and manually created templates for them, then add them to our *template bank*. It is important to note that for each property, we only need to predefine templates the first time our system encounters this property. For instance, if a concept has the property "country", we define templates for this property just once and add to the bank. Subsequent concepts with the same property can then reuse these templates from the bank. That means the number of templates to be hand-crafted will gradually decrease until all potential properties have corresponding templates in our bank, at which point the question generation process becomes completely automatic.

Specifically, for each property, we define a *main template* and an optional *sub-clause template*. Our template bank consists of 182 main templates and 100 sub-clause templates for 182 distinct properties. These numbers can increase as our framework can be extended to include additional image sources and knowledge bases. Table 1 presents a few examples in our template bank.

Property	Templates
architect	(a) `Who designed __ ?` (b) `the architect of __`
author	(a) `Who created __ ?` (b) `the author of __`
country	(a) `In which country is __ located?` (b) `where __ is located`
height	(a) `How high is __?`
width	(a) `How wide is __?`
official language	(a) `What is the official language of __?` (b) `the official language of __`
currency	(a) `What is the currency of __?` (b) `the currency of __`
capital	(a) `What is the capital of __?` (b) `the capital of __`
mother	(a) `Who is the mother of __?` (b) `the mother of __`
place of birth	(a) `Where was __ born?` (b) `the place of birth of __`

Table 1. Examples of predefined templates. For each property, we define a (a) *main template* and an optional (b) *sub-clause template*.

1.3. Answer Distribution Balancing

To reduce the bias in the answer distribution, we iteratively apply a balancing process following three criteria: (1) preserving the relative size of *head* and *tail*; (2) maintaining the frequency order; and (3) prioritizing the removal of answers associated with a higher number of questions. The *head* represents the group of questions with the most answers, while the *tail* represents the group with the least. Figure 3 illustrates an example of the answer distribution before and after applying the balancing process for 10 and 20 rounds. Questions and answers are distributed into groups based on the properties from which they were generated. While the balancing process applies to all groups, we visualize only the top 20 groups with the highest number of answers. For each group, we also visualize only the top 10 most frequent answers, in descending order. After 20 iterations, 26,100 questions were discarded, which is 33.4% of the total number of questions. The answer distribution became much more balanced, with a few groups on the left side showing the most noticeable improvement.

1.4. Dataset Splitting

Figure 4 shows the similarity in answer distribution between the train set and test set. Here we also visualize top 10 most frequent answers from the top 20 groups.

2. Dataset Analysis

In this section, we present more statistics of our dataset and provide details of our user study conducted for question quality evaluation.

2.1. Dataset Statistics and Examples

The latest version of ReasonVQA consists of nearly 4.2M generated from 598K images, with 1.3M 1-hop questions, 2.8M 2-hop questions, and 5.4K 3-hop questions. Our dataset statistics are shown in Table 2. Figure 5 illustrates the distribution of questions by the first four words. Figure 6 presents multiple instances from ReasonVQA.

	ReasonVQA	**ReasonVQA-U**
# Images	598,525	13,326
# Questions	4,174,024	78,007
# 1-hop questions	1,358,634	23,767
# 2-hop questions	2,809,960	49,459
# 3-hop questions	5,430	4,781
# Unique questions	123,204	22,368
# Unique answers	73,068	9,103
# Unique choices	123,411	21,037
Avg. question length (words)	9.77	9.62
Avg. answer length (words)	1.53	1.49

Table 2. Some characteristics of our datasets in full version and subset version.

2.2. Dataset Domains

As detailed in the main paper, we categorized questions into 20 domains, outlined as follows. Figure 7 visualizes the domain distribution.

1. **Places & Locations**
   e.g. Country where a place is located
2. **Person & Institutions**
   e.g. Organization employing an individual
3. **Temporal Concepts**
   e.g. Official opening date
4. **Characteristics & Properties**
   e.g. Height of buildings or structures
5. **Language & Cultural**
   e.g. Language officially recognized in a region
6. **History & Events**
   e.g. Date or people associated with a historical event
7. **Physical Geography**
   e.g. Capital city of a country
8. **Politics & Ideologies**
   e.g. Head of government
9. **Economics & Labor**
   e.g. Industry associated with an organization
10. **Nature & Human Interaction**
    e.g. Water composition of a given area
11. **Technology & Innovation**
    e.g. Manufacturer of a technological item
12. **Science & Quantitative Analysis**
    e.g. Temperature or light range of an object
13. **Health & Medicine**
    e.g. Symptoms associated with a condition
14. **Education & Knowledge Systems**
    e.g. Institution where an individual was educated
15. **Art & Creative Expressions**
    e.g. Collection housing an artistic work
16. **Philosophy & Spiritual Beliefs**
    e.g. Entity or concept to which a church is dedicated
17. **Media & Communication Systems**
    e.g. Number of episodes in a series
18. **Environment & Sustainability**
    e.g. Inflow and outflow of lakes
19. **Law & Justice Systems**
    e.g. Area of legal authority
20. **Food & Nutrition**
    e.g. Caloric content of food or drink

2.3. Question Evaluation by User Study

To evaluate the quality of generated questions, we conducted a user study with 1,000 randomly selected question and image pairs. Twenty participants, all proficient in English as their primary language for work or study, assessed the correctness of the answers and the naturalness of 50 randomly chosen questions each. For the naturalness, they rated the questions on a four-level scale: (1) very unnatural, (2) unnatural, (3) natural, and (4) very natural. Additionally, they were also asked to mark any questions with grammatical errors. The results indicated that 96% of selected answers are correct, 2.2% of the questions were rated as "very

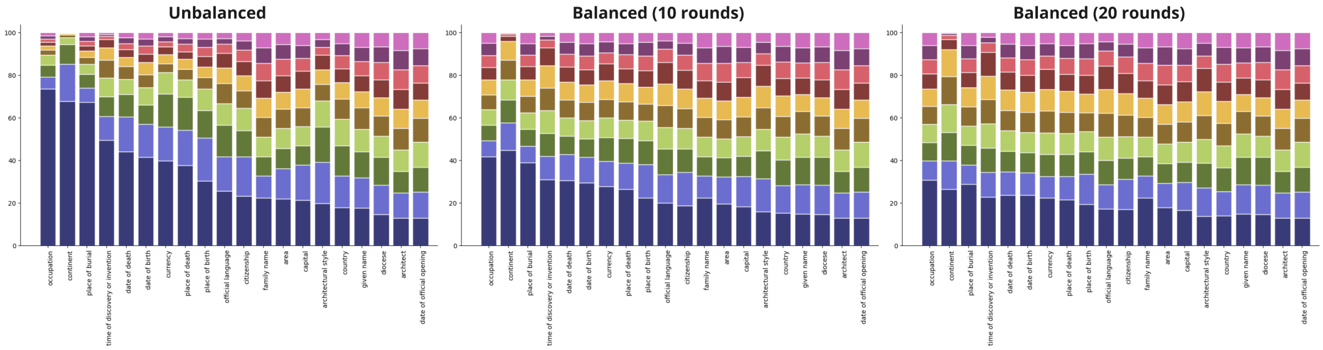

Figure 3. The answer distribution for the unbalanced dataset (left) and the balanced datasets after 10 rounds (middle) and 20 rounds (right) of the balancing process. We show the top 10 answers in the top 20 groups. The column height corresponds to the relative frequency of each answer. The distribution started from being heavily biased to becoming more uniform.

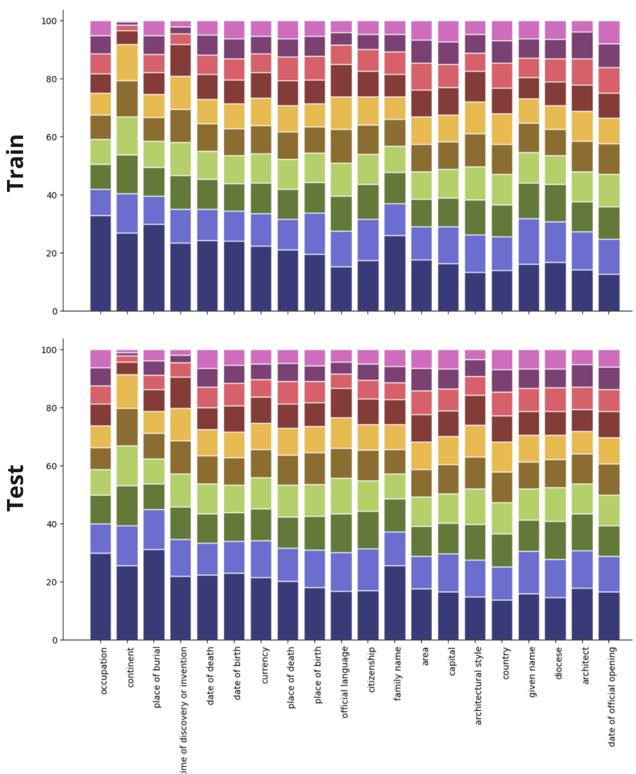

Figure 4. The answer distribution of the train and test set. The same top 10 answers in the top 20 groups for the train set (top) and test set (bottom). The column height corresponds to the relative frequency of each answer.

Figure 5. Distribution of questions by first four words

unnatural," 13.9% as "unnatural," 58.1% as "natural," and 25.8% as "very natural,", with only 2.5% of the questions had grammatical errors. The results are shown in Figure 8.

3. Experiments

In this section, we provide experimental details regarding different dataset sizes. We also present the average accuracies across domains, along with additional metrics for the benchmarked models, including the Standard Deviation (SD) and the Standard Error of the Mean (SEM). These scores are further analyzed across various aspects of our dataset.

3.1. Benchmark Results Across Dataset Sizes

To understand how models perform with varying dataset sizes, we conducted experiments on seven models Mantis-SigLIP [5], Mantis-Idefics2 [5], mPLUG-Owl3 [15], GPT-4o [4], LLaVA-OV [8], Qwen2.5-VL [1], PaliGemma-2-Mix [12] across a range of dataset sizes. In our experiments, we aimed to investigate how model performance varies with dataset size while controlling for sample difficulty. To this end, we first divided the full dataset into three pools based

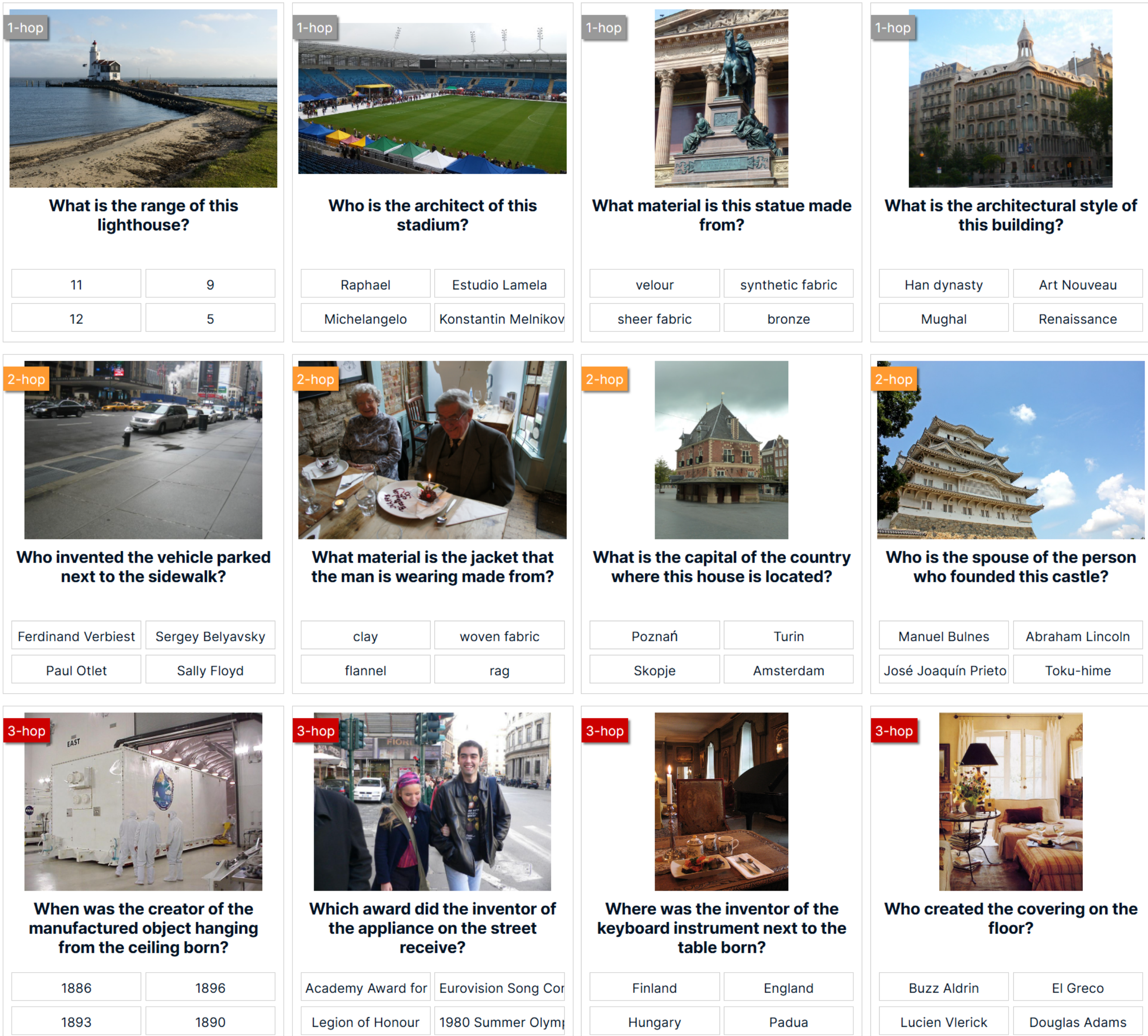

Figure 6. Some example questions and answers from ReasonVQA. The first row shows 1-hop questions, the middle row shows 2-hop questions with the first two questions constructed by incorporating the scene graph, and the last row contains 3-hop questions.

on the number of hops. For each target dataset size (e.g., $10k$, $20k$, …, $80k$ samples), we randomly sampled examples from each pool proportionally, maintaining the same ratio of as in the full dataset. This strategy ensures that each subset reflects the overall difficulty distribution, enabling us to evaluate how model performance scales with dataset size without bias toward easier or harder samples. Figure 9 illustrates the performance of various models across different dataset sizes. We observe that as the dataset grows, model performance initially improves but then declines with varying degrees of intensity. This decline likely occurs because as new samples are introduced, they bring additional domain knowledge and more complex questions, which may challenge the ability of models to maintain efficiency and accuracy under the increased data load. This observation suggests that integrating a wider variety of image sources and knowledge domains could create a more demanding benchmark, providing a more comprehensive assessment of model robustness.

3.2. Benchmark Results Across Dataset Aspects

For a fine-grained analysis of our dataset, we evaluated the accuracy of 13 large language models (LLMs) (BLIP-2 [9], InstructBLIP [3], mPLUG-Owl2 [16], Idefics2 [7], Mantis-

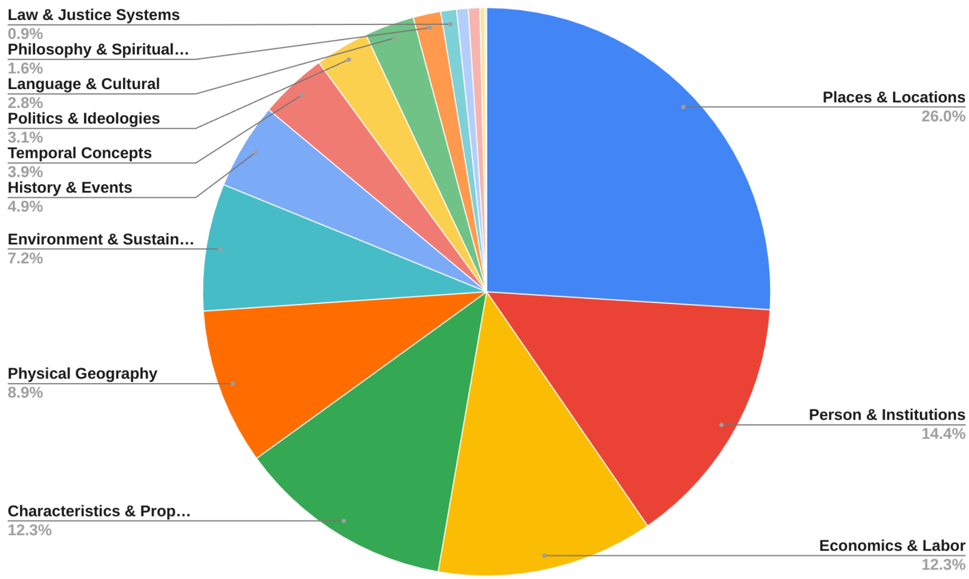

Figure 7. Distribution of questions across domains. Each question is categorized into specific domains based on the property from which it was generated.

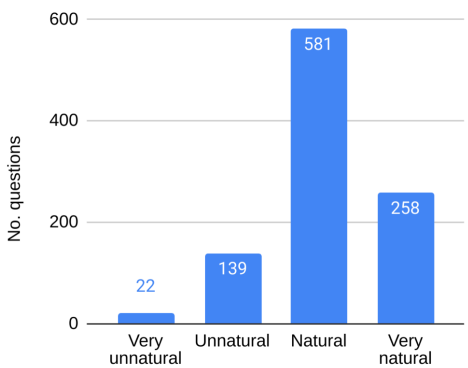

Figure 8. Visualization of the user study for question naturalness evaluation.

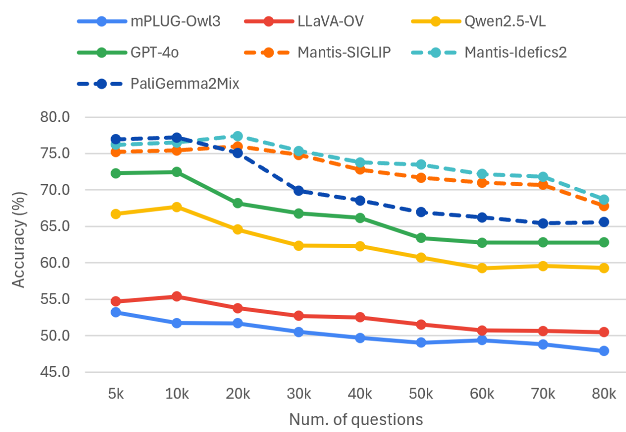

Figure 9. Performance of models across different dataset sizes in open-ended (solid lines) and multiple choice (dashed lines) scenarios, with accuracy reported.

SigLIP [5], Mantis-Idefics2 [5], mPLUG-Owl3 [15], GPT-4o [4], LLaVA-OV [8], Qwen2.5-VL [1], PaliGemma-2 [13], PaliGemma-2-Mix [12], SmolVLM-Instruct [10]) across all domains, as shown in Figure 10. We observed that Health & Medicine proved to be the most challenging domain, achieving an average accuracy of only 20.7%. This was closely followed by Art & Creative Expressions, with an average accuracy of 22.5%. The most difficult question types consistently involved numerical values, such as inquiries about the area of a city. Furthermore, we computed Standard Error of the Mean (SEM) and Standard Deviation (SD) scores to better illustrate the variability within our dataset, as presented in Table 3 and 4. Overall, the SD values are relatively large, indicating a high degree of variability in the model performances across different aspects of our dataset. This variability indicates that ReasonVQA poses significant challenges to the models, as large SD values typically reflect a high sensitivity to the diverse types of samples or tasks within the dataset. At the same time, the relatively low SEM values (ranging from 0.1% to 0.7%) suggest that the mean accuracy scores for each model are estimated with high precision, despite the substantial variability indicated by the SD. This highlights the presence of considerable individual sample variability, with the models demonstrating inconsistent performance across different subsets of the dataset.

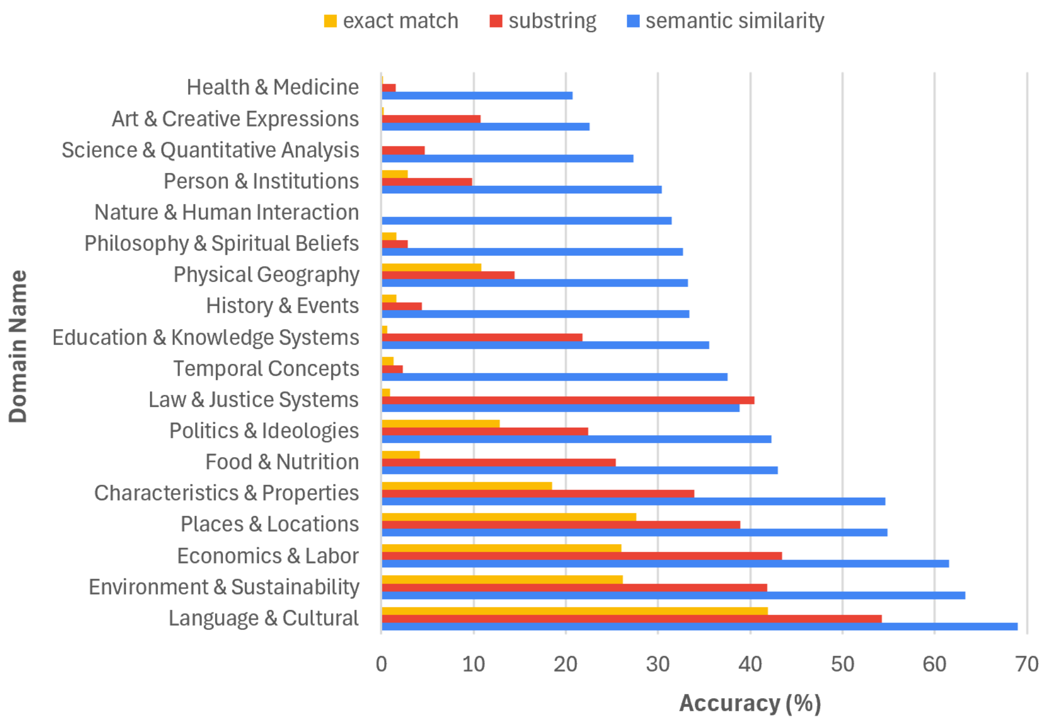

Figure 10. Average accuracy of models across domains.

Model	Overall	1-hop	2-hop	3-hop	SG	no SG	VG	GLDv2
BLIP-2	**31.3** (0.1)	**31.3** (0.2)	**31.5** (0.1)	**16.3** (0.2)	**16.6** (0.2)	**32.1** (0.1)	**16.6** (0.2)	**32.1** (0.1)
InstructBLIP	**30.9** (0.1)	**28.6** (0.2)	**31.7** (0.1)	**16.1** (0.2)	**18.7** (0.2)	**31.6** (0.1)	**18.7** (0.2)	**31.6** (0.1)
mPLUG-Owl2	**17.2** (0.1)	**17.6** (0.1)	**17.1** (0.1)	**9.9** (0.1)	**10.6** (0.1)	**17.6** (0.1)	**10.6** (0.1)	**17.6** (0.1)
Idefics2	**31.4** (0.1)	**28.5** (0.2)	**33.2** (0.1)	**18.0** (0.3)	**20.4** (0.2)	**32.0** (0.1)	**20.4** (0.2)	**32.0** (0.1)
Mantis-SigLIP	**26.1** (0.1)	**23.4** (0.2)	**28.0** (0.1)	**17.2** (0.2)	**18.3** (0.2)	**26.9** (0.1)	**18.3** (0.2)	**26.9** (0.1)
Mantis-Idefics2	**31.0** (0.1)	**26.5** (0.2)	**33.7** (0.2)	**18.9** (0.3)	**20.5** (0.2)	**31.8** (0.1)	**20.5** (0.2)	**31.8** (0.1)
mPLUG-Owl3	**29.4** (0.1)	**26.2** (0.2)	**31.3** (0.1)	**17.2** (0.2)	**20.4** (0.2)	**30.0** (0.1)	**20.4** (0.2)	**30.0** (0.1)
LLaVA-OV	**30.7** (0.1)	**29.2** (0.2)	**31.8** (0.1)	**18.4** (0.3)	**18.4** (0.2)	**31.3** (0.1)	**18.4** (0.2)	**31.3** (0.1)
Qwen2.5-VL	**33.4** (0.1)	**32.5** (0.2)	**33.8** (0.2)	**17.2** (0.2)	**20.6** (0.2)	**33.7** (0.1)	**20.6** (0.2)	**33.7** (0.1)
GPT-4o	**33.0** (0.1)	**31.0** (0.2)	**33.0** (0.2)	**17.8** (0.3)	**21.2** (0.2)	**32.4** (0.1)	**21.2** (0.2)	**32.4** (0.1)
PaliGemma-2	**24.7** (0.1)	**22.5** (0.2)	**26.1** (0.1)	**12.1** (0.2)	**11.8** (0.1)	**25.3** (0.1)	**11.8** (0.1)	**25.3** (0.1)
PaliGemma-2-Mix	**31.6** (0.1)	**30.0** (0.2)	**32.9** (0.1)	**18.9** (0.3)	**19.3** (0.2)	**32.4** (0.1)	**19.3** (0.2)	**32.4** (0.1)
SmolVLM	**15.7** (0.1)	**15.5** (0.2)	**15.3** (0.1)	**8.7** (0.2)	**10.8** (0.2)	**15.8** (0.1)	**10.8** (0.2)	**15.8** (0.1)

Table 3. SD and SEM scores across various models on our datasets in the zero-shot open-ended scenario. The accuracies are computed using the *semantic similarity* string-matching method. SD scores are highlighted in bold, with SEM scores provided in parentheses. “SG” refers to scene graph. “VG” and “GLDv2” refer to Visual Genome and Google Landmarks Datasets v2 respectively.

Model	Overall	1-hop	2-hop	3-hop	SG	no SG	VG	GLDv2
BLIP-2	**47.9** (0.2)	**43.7** (0.3)	**48.4** (0.2)	**46.5** (0.7)	**48.9** (0.5)	**47.0** (0.2)	**48.9** (0.5)	**47.0** (0.2)
InstructBLIP	**47.6** (0.2)	**44.7** (0.3)	**47.4** (0.2)	**43.8** (0.6)	**48.6** (0.5)	**46.4** (0.2)	**48.6** (0.5)	**46.4** (0.2)
mPLUG-Owl2	**47.5** (0.2)	**44.6** (0.3)	**47.8** (0.2)	**48.6** (0.7)	**49.8** (0.5)	**46.7** (0.2)	**49.8** (0.5)	**46.7** (0.2)
Idefics2	**46.4** (0.2)	**41.7** (0.3)	**46.5** (0.2)	**45.2** (0.7)	**49.2** (0.5)	**44.8** (0.2)	**49.2** (0.5)	**44.8** (0.2)
Mantis-SigLIP	**46.7** (0.2)	**42.8** (0.3)	**46.7** (0.2)	**44.0** (0.6)	**48.9** (0.5)	**45.2** (0.2)	**48.9** (0.5)	**45.2** (0.2)
Mantis-Idefics2	**46.4** (0.2)	**41.9** (0.3)	**46.3** (0.2)	**42.3** (0.6)	**48.4** (0.5)	**44.7** (0.2)	**48.4** (0.5)	**44.7** (0.2)
mPLUG-Owl3	**46.3** (0.2)	**42.6** (0.3)	**46.3** (0.2)	**47.4** (0.7)	**49.7** (0.5)	**45.0** (0.2)	**49.7** (0.5)	**45.0** (0.2)
LLaVA-OV	**49.5** (0.2)	**48.3** (0.3)	**49.7** (0.2)	**48.2** (0.7)	**49.5** (0.5)	**49.3** (0.2)	**49.5** (0.5)	**49.3** (0.2)
PaliGemma-2	**35.4** (0.1)	**40.6** (0.3)	**32.5** (0.1)	**33.0** (0.5)	**33.1** (0.4)	**35.6** (0.1)	**33.1** (0.4)	**35.6** (0.1)
PaliGemma-2-Mix	**47.5** (0.2)	**42.0** (0.3)	**48.0** (0.2)	**44.1** (0.7)	**47.1** (0.5)	**46.1** (0.2)	**47.1** (0.5)	**46.1** (0.2)

Table 4. SD and SEM scores across various models on our datasets in the zero-shot multiple choice scenario. SD scores are highlighted in bold, with SEM scores provided in parentheses. “SG” refers to scene graph. “VG” and “GLDv2” refer to Visual Genome and Google Landmarks Datasets v2 respectively.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Ming-Hsuan Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *CoRR*, abs/2502.13923, 2025. 4, 6

[2] Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. O’Reilly, 2009. 1

[3] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven C. H. Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 5

[4] Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Madry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll L. Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, and Dane Sherburn. Gpt-4o system card. *CoRR*, abs/2410.21276, 2024. 4, 6

[5] Dongfu Jiang, Xuan He, Huaye Zeng, Cong Wei, Max Ku, Qian Liu, and Wenhu Chen. Mantis: Interleaved multi-image instruction tuning. *Trans. Mach. Learn. Res.*, 2024, 2024. 4, 6

[6] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *Int. J. Comput. Vis.*, 123(1):32–73, 2017. 1

[7] Hugo Laurençon, Léo Tronchon, Matthieu Cord, and Victor Sanh. What matters when building vision-language models? In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. 5

[8] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *Trans. Mach. Learn. Res.*, 2025, 2025. 4, 6

[9] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, pages 19730–19742. PMLR, 2023. 5

[10] Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben

Allal, Anton Lozhkov, Nouamane Tazi, Vaibhav Srivastav, Joshua Lochner, Hugo Larcher, Mathieu Morlon, Lewis Tunstall, Leandro von Werra, and Thomas Wolf. Smolvlm: Redefining small and efficient multimodal models. *arXiv preprint arXiv:2504.05299*, 2025. 6

[11] George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, 1995. 1

[12] Merve Noyan, Aritra Roy Gosthipaty, and Andreas P. Steiner. Paligemma 2 mix - new instruction vision language models by google. *arXiv preprint arXiv:2412.03555*, 2025. 4, 6

[13] Andreas Steiner, André Susano Pinto, Michael Tschannen, Daniel Keysers, Xiao Wang, Yonatan Bitton, Alexey A. Gritsenko, Matthias Minderer, Anthony Sherbondy, Shangbang Long, Siyang Qin, R. Reeve Ingle, Emanuele Bugliarello, Sahar Kazemzadeh, Thomas Mesnard, Ibrahim Alabdulmohsin, Lucas Beyer, and Xiaohua Zhai. Paligemma 2: A family of versatile vlms for transfer. *CoRR*, abs/2412.03555, 2024. 6

[14] Tobias Weyand, André Araújo, Bingyi Cao, and Jack Sim. Google landmarks dataset v2 - A large-scale benchmark for instance-level recognition and retrieval. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 2572–2581. Computer Vision Foundation / IEEE, 2020. 1

[15] Jiabo Ye, Haiyang Xu, Haowei Liu, Anwen Hu, Ming Yan, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mplug-owl3: Towards long image-sequence understanding in multimodal large language models. *CoRR*, abs/2408.04840, 2024. 4, 6

[16] Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. *CoRR*, abs/2311.04257, 2023. 5