# Table and Image Generation for Investigating Knowledge of Entities in Pre-trained Vision and Language Models

**Hidetaka Kamigaito**[†], **Katsuhiko Hayashi**[‡], **Taro Watanabe**[†]
[†]Nara Institute of Science and Technology [‡]Hokkaido University
{kamigaito.h, taro}@is.naist.jp
katsuhiko-h@ist.hokudai.ac.jp

## Abstract

In this paper, we propose a table and image generation task to verify how the knowledge about entities acquired from natural language is retained in Vision & Language (V&L) models. This task consists of two parts: the first is to generate a table containing knowledge about an entity and its related image, and the second is to generate an image from an entity with a caption and a table containing related knowledge of the entity. In both tasks, the model must know the entities used to perform the generation properly. We created the Wikipedia Table and Image Generation (WikiTIG) dataset from about 200,000 infoboxes in English Wikipedia articles to perform the proposed tasks. We evaluated the performance on the tasks with respect to the above research question using the V&L model OFA (Wang et al., 2022), which has achieved state-of-the-art results in multiple tasks. Experimental results show that OFA forgets part of its entity knowledge by pre-training as a complement to improve the performance of image related tasks.

## 1 Introduction

Vision & Language (V&L), which is the fusion of vision and language tasks, has achieved great success in tasks such as caption generation from images (Xu et al., 2015) and image generation from texts (Reed et al., 2016). This progress has been driven by pre-trained V&L models that are trained on large-scale V&L datasets (Du et al., 2022). To generate appropriate captions and images for input, pre-trained V&L models need to have prior knowledge of the features of the objects they are generating (Cao et al., 2020; Yun et al., 2021). These models retain knowledge about entities in particular by inheriting parameters from pre-trained language models used in natural language processing to indirectly utilize data resources such as Wikipedia.

In this way, V&L models (Lu et al., 2019; Su et al., 2020; Li et al., 2020; Cho et al., 2021; Wang et al., 2022; Saharia et al., 2022) map the inherited textual knowledge into visual representations through additional training on V&L datasets.

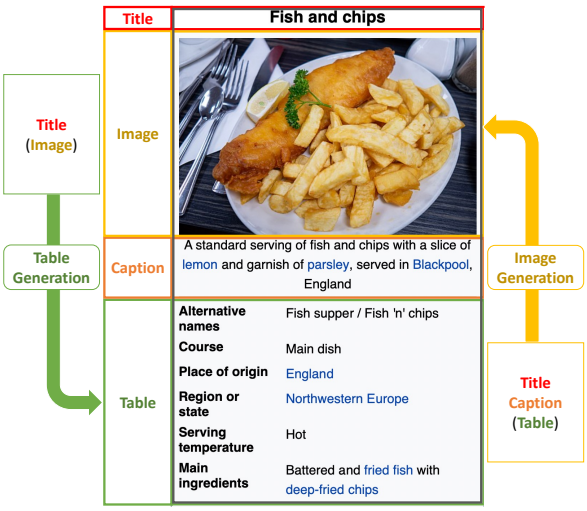



Figure 1: An infobox of a Wikipedia article[1]. In this study, we validate the V&L model by generating images and tables in infoboxes.

This learning process raises a number of questions, such as whether the knowledge about entities acquired from natural language is adequately retained in the pre-trained V&L model, or whether it is enhanced by combining it with image features. These are important in understanding the limits of what can be generated by the pre-trained V&L model.

To answer these questions, we propose a task of generating tables and images of infoboxes in English Wikipedia. Figure 1 shows an example of the target infobox, in which either tables or images are generated by the proposed task. In both cases, the model must know the entities to generate them properly.

We collected about 200,000 infoboxes to construct the Wikipedia Table and Image Generation (WikiTIG) dataset necessary to perform the pro-

[1] https://en.wikipedia.org/wiki/Fish_and_chips

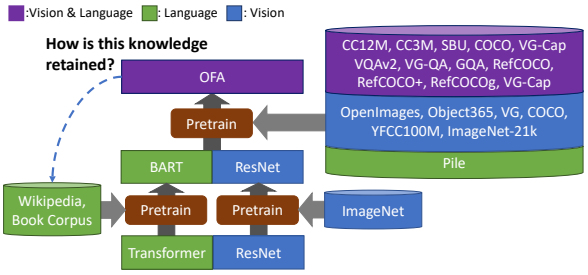



Figure 2: Learning process of OFA. We investigate how OFA retains knowledge about entities acquired from pre-training on Wikipedia articles.

posed task. In addition, we used OFA (Wang et al., 2022), a pre-trained V&L model that has achieved state-of-the-art performance in various V&L tasks.

Our evaluation of the table generation revealed that part of the knowledge in the V&L model acquired from natural language is lost when the V&L model is pre-trained. We also found that additional knowledge for entities was acquired by supplementing image information, which was not possible solely from textual data.

In image generation, we found that OFA can generate more accurate images by using the knowledge expressed in the table. We also found that the models trained only on natural language can infer table knowledge, which increases the diversity of generated images. Our code and dataset will be released at https://github.com/kamigaito/WikiTIG.

## 2 Vision & Language Models

Many pre-trained V&L models have achieved state-of-the-art performance on various tasks by inheriting the weights of the conventional pre-trained models for natural language and images (Lu et al., 2019; Su et al., 2020; Li et al., 2020; Cho et al., 2021; Wang et al., 2022; Saharia et al., 2022) before learning V&L datasets. Our study examines how the knowledge represented in the pre-trained model for natural language is transformed through such a learning process. We select OFA, which has achieved state-of-the-art performance in multiple V&L tasks, as our target model.

Figure 2 shows the network structure of OFA and its relation to each dataset[2]. OFA uses VQGAN (Esser et al., 2020) on the decoder to transform images into discrete sequences so that the same Transformer (Vaswani et al., 2017) is used for image and natural language generation. Because OFA inherits

[2]Appendix A describes the data for the pre-training.

| Task | Input | Output |
|---|---|---|
| Table Generation | Title, Image | Table |
| Image Generation | Title, Caption, Table | Image |

Table 1: Outline of each task. See Figure 1 for the parts of the infobox to which each term refers.

Alternative names | Fish supper / Fish 'n' chips <> Course | Main dish <> Place of origin | England <> Region or state | Northwestern Europe <> Serving temperature | Hot <> Main ingredients | Battered and fried fish with deep-fried chips

Figure 3: This example is a linearized version of the table in Figure 1.

parameters from BART (Lewis et al., 2020), which shares a similar Transformer structure, OFA should include knowledge acquired from natural language such as Wikipedia articles. Unlike the decoder, the encoder handles images directly; thus, OFA uses the output of ResNet (He et al., 2016) to embed images in addition to the embedding layer inherited from BART.

## 3 Table and Image Generation

In this section, we describe two tasks for verifying knowledge behavior in the V&L model: table generation and image generation. Both tasks are based on infoboxes in Wikipedia articles, which correspond to summary information of the Wikipedia articles comprising tables and images[3]. Thus, it is suitable for verifying the knowledge about entities in Wikipedia kept in the pre-trained V&L model. In the following subsections, we explain the details of each task.

### 3.1 Table Generation

In the table generation task, the target V&L model generates a table from a title and/or image of the infobox. To do this, the model generates linearized tables, similarly to table generation by descriptions (Wu et al., 2022b). In our setting, we linearize tables as shown in Figure 3 using the column separator "|" and the row separator "<>" to reuse pre-trained token embeddings. The separator symbols are accompanied by spaces before and after for use in BPE tokenization. We investigate the target model by directly generating such linearized text. We use the following settings for the investigation.

[3]https://en.wikipedia.org/wiki/Help:Infobox

**Generation from titles** We investigate the knowledge about entities held by V&L models by comparing tables generated from titles by pre-trained V&L models and by pre-trained models trained only on natural language.

**Generation from title and images** We generate tables from titles with images and compare the results with those generated from only titles. This enables us to investigate the new knowledge in pre-trained V&L models transferred from images.

**Metrics** For comparison, we use the following evaluation metrics to measure how close the generated tables are to the actual ones.

- **ROUGE**: Since the linearized tables are text data and the infobox plays the role of summarizing the article, we use ROUGE (Lin, 2004), the most widely used evaluation method for automatic summarization. In our evaluation with ROUGE, we convert the column separator "|" and the row separator "<>" to spaces so that the sequence of strings is not restricted to rows and columns.
- **Table-F$_1$**: To evaluate the tables with respect to their structure, we divide the cells by their types and then evaluate the matches with the reference table in terms of the $F_1$ measure for each case and average them. When calculating the matches, we apply clipping used in ROUGE to prevent the score from increasing due to the repetition of the same cell in the output[4]. We treat cells of each type separately[5] as follows:

  - **Group**: The infobox sometimes divides the table into groups, with the first row of each group serving as a header for the group name. The prediction performance for the group names is important for verifying what aspects of knowledge the model has about the entities. Since these rows consist of a single column, we target rows consisting of a single column in this type of cell.

  - **Header**: The head of each row in the table consisting of more than one column is usually the header of a subsequent cell in the same row. Therefore, the prediction performance for headers is important for the same reason as for group names.

  - **Value**: The second cells in each row of a table with two columns have values corresponding to the headers. Therefore, the prediction performance of the values is important for knowing whether the model has detailed knowledge about the entity. To examine the correspondence between headers and their values, we treat a header and its corresponding value as a pair.

- **Corpus-F$_1$**: Because the above Table-$F_1$ computes each case individually, it is difficult to evaluate how much diverse knowledge the model outputs. To solve this problem, we share cells across all instances and compute $F_1$ values in a batch. Similarly to Table-$F_1$, we apply clipping to the score calculation[6] and treat cell types Group, Header, and Value separately as defined in Table-$F_1$.

[4]Appendix B.1 shows the details of this calculation.
[5]Appendix C shows an example of the cell types.

| Task | Total | Train | Valid | Test |
|---|---|---|---|---|
| Table Generation | 204,460 | 184,124 | 10,081 | 10,255 |
| Image Generation | 86,654 | 78,012 | 4,261 | 4,381 |

Table 2: The data size for each task in the WikiTIG dataset.

### 3.2 Image Generation

In the image generation task, the model receives a title, caption, and table to generate the corresponding image:

**Generation from a title and caption** By using the minimum input required to generate images, we investigate the difficulty of generating them compared to other datasets.

**Generation from a title, caption, and table** We investigate the impact of knowledge about entities on image generation by generating images from input, including tables, and compare the results to the setting without tables.

**Metrics** We use the following three widely used measures for evaluating image generation.

- **CLIP:** The relevance of the input text to the generated images inferred by the pre-trained V&L model CLIP (Radford et al., 2021).
- **Inception Score (IS)**: How easily a model can distinguish the differences between each image and the variety of generated images (Salimans et al., 2016). It is inferred by the pre-trained image classification model Inception-v3 (Szegedy et al., 2016).
- **Frechet Inception Distance (FID)**: How close the generated image is to the reference image, es-

[6]Appendix B.2 shows the details of this calculation.

| Model | Input | ROUGE ↑ | | | Table-$F_1$ ↑ | | | Corpus-$F_1$ ↑ | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | L | Header | Group | Value | Header | Group | Value |
| BART | Title | 28.8±0.2 | 14.0±0.1 | 26.6±0.1 | 38.9±0.1 | **24.3**±0.1 | **4.9**±0.0 | **62.9**±0.3 | **35.5**±0.0 | **11.7**±0.0 |
| OFA | Title | 28.1±0.2 | 13.4±0.1 | 25.7±0.2 | 34.7±0.4 | 22.8±0.2 | 4.3±0.1 | 57.8±0.7 | 33.3±0.2 | 10.7±0.2 |
| OFA | Image | 28.0±0.1 | 11.5±0.0 | 25.8±0.1 | 41.9±0.1 | 21.2±0.1 | 2.7±0.0 | 57.4±0.2 | 26.6±0.2 | 6.8±0.0 |
| OFA | Both | **31.3**±0.1 | **14.2**±0.1 | **28.7**±0.1 | **43.5**±0.1 | 23.2±0.1 | 3.7±0.0 | 59.2±0.2 | 28.6±0.1 | 8.2±0.1 |

Table 3: Table generation results. Bold font denotes the highest score, and ↑ denotes that the higher the score, the more optimal. ± denotes the standard deviation of the score. *Both* means the input contains both a title and image. Underline indicates that the score improvement is statistically significant from the second-highest one ($p < 0.05$)[7].

timated by Inception-v3 like IS. A lower FID is more ideal.

## 4 Dataset Creation

We created the Wikipedia Table and Image Generation (WikiTIG) dataset by extracting infoboxes from the HTML dump data of the English Wikipedia[8]. To ensure consistency in the format of infoboxes, we limited the extraction target to those containing a title in the first row and an image in the second row, as shown in Figure 1.

In order to use only entities with sufficient information, we targeted entities for which the table was not empty. In addition, to ensure reliable correspondence, only rows one column wide, which often describe groups, and rows two columns wide, which often consist of a header and its value, were targeted for extraction.

The target images are limited to those in jpeg, png, and gif formats. Since some captions do not include a title, we used a hyphen to join the title at the beginning of the caption in such cases.

Table 2 shows the size of each dataset. The dataset size diverges between two tasks because some infoboxes do not include captions[9].

## 5 Evaluation & Analysis

### 5.1 Table Generation

**Settings** We chose OFA (Wang et al., 2022), a pre-trained V&L model, and BART (Lewis et al., 2020), pre-trained only in natural language, as models for comparison. For both models, we used the `base` settings with the hyperparameters reported in Wang et al. (2022). We performed the training three times with different seeds and reported their average scores with their standard deviations[10].

**Results** Table 3 shows the results for each setting in the table generation[11]. When only the title is used as input, the result of BART is more accurate than that of OFA, indicating that part of the knowledge acquired from natural language is lost due to additional learning in the V&L model. The use of image information improves Table-$F_1$ for headers, indicating that images reinforce the knowledge of what kind of features an entity has.

In contrast, $F_1$ for cell values did not improve, indicating that information obtained from images does not complement detailed knowledge, such as the values corresponding to each header obtained from natural language.

The results of BART in Corpus-$F_1$ also suggest that BART contains more diverse knowledge internally than in other settings. This result reinforces that the V&L model forgot part of the knowledge from natural language through additional learning, and images could not fully complement them.

### 5.2 Image Generation

**Settings** Similarly to the table generation, we chose OFA for the comparison. We additionally join the reference tables (Gold) and those generated by models in §5.1 (OFA, BART) as the input in order to investigate the impact of the ability to infer table knowledge. We also used the `base` settings with the hyperparameters reported in Wang et al. (2022). We also performed the training three times with different seeds and reported their average scores with their standard deviations[12].

**Results** Table 4 shows the results for each setting in the image generation[13]. Since the CLIP value

[7] We used paired-bootstrap resampling (Koehn, 2004) for the significance test.

[8] `https://dumps.wikimedia.org/other/static_html_dumps/current/en/` (CC BY-SA 3.0).

[9] See Appendix D for the dataset details.

[10] See Appendix E.1 for the detailed settings.

[11] Appendix F.1 shows the generated images.

[12] See Appendix E.2 for the detailed settings.

[13] Appendix F.2 shows the generated images.

| Input | CLIP ↑ | IS ↑ | FID ↓ |
|---|---|---|---|
| Title & Caption | 28.7±0.0 | 10.5±0.1 | 31.1±0.2 |
| +Table (Gold) | **29.4**±0.0 | **11.3**±0.2 | **28.5**±0.3 |
| +Table (BART) | 28.1±0.0 | 10.6±0.2 | 32.4±0.3 |
| +Table (OFA) | 28.0±0.1 | 10.6±0.2 | 33.1±0.4 |

Table 4: Image generation results. ↓ denotes that the lower the score, the more optimal the result. + denotes additionally used input to the title and captions. The parenthesis denotes the origin of the table. Other notations are the same as in Table 3.

in OFA is close to the result (Wang et al., 2022) in MS COCO (Chen et al., 2015) for image generation, the use of our created dataset is reasonable for training models. In addition, the input of Table (Gold) improves all metrics, indicating that the model produces higher quality images when provided with complementary knowledge about the entities. This result also indicates that OFA does not retain sufficient knowledge of the entities in English Wikipedia.

In addition, we did not observe any performance improvement in CLIP and FID when fed with automatically generated tables from BART and OFA. However, tables generated by BART improves IS with the lower performance degradation of FID than that by OFA, indicating that automatically generated tables can improve the diversity of the output images and accurate tables are more important for improving performance in image generation.

## 6 Related Work

Following the advancements in V&L models (Du et al., 2022), there have been various studies that investigate V&L models. Cao et al. (2020) conducted a comprehensive analysis of V&L models including the difference between model structures. Through their analysis, they revealed the importance of text information in V&L tasks over image information.

Several studies focused on the performance differences between V&L models and text-only models. Yun et al. (2021) investigated the improvement of linguistic representations by pre-training V&L models on PhysicalQA (PIQA) (Bisk et al., 2020) and the probing framework of (Tenney et al., 2019). They concluded that the benefit of pre-trained V&L models for text-only tasks is marginal. Iki and Aizawa (2021); Hagström and Johansson (2022) compared the performance of V&L models and text-only models on the text-only benchmark, GLUE (Wang et al., 2018) and determined that the text-only model achieved higher scores than the V&L models.

However, even though various kinds of V&L models (Lu et al., 2019; Su et al., 2020; Li et al., 2020; Cho et al., 2021; Wang et al., 2022; Saharia et al., 2022) inherit language-related knowledge from pre-trained language-only models, how the knowledge is inherited has yet to be investigated. Our work clarifies this by using our created dataset, Wikipedia Table and Image Generation (WikiTIG).

## 7 Conclusion

This paper investigates how knowledge about entities are preserved in a pre-trained V&L model which is originally transferred from a pre-trained natural language model.

We analyzed a pre-trained V&L model by creating the Wikipedia Table and Image Generation (WikiTIG) dataset for generating images and tables of the infoboxes in Wikipedia. WikiTIG consists of 200,000 infoboxes and their corresponding images from English Wikipedia.

Experimental results on a pre-trained V&L model OFA (Wang et al., 2022) showed that the model forgot part of the knowledge about entities during pre-training, and the image information did not fully compensate for the forgotten knowledge.

## Limitations

Regarding the Wikipedia articles used for creating our dataset Wikipedia Table and Image Generation (WikiTIG), some infoboxes may not follow the defined format and rules. This is because various users can freely edit infoboxes. Moreover, the HTML dump data published by English Wikipedia is not based on recent information.

In image generation, due to the standard settings recommended by Zhang et al. (2021); Ramesh et al. (2021); Wang et al. (2022); Wu et al. (2022a), our image generation task requires generating a cropped fixed-size square image instead of the original aspect ratio.

In addition, a table in an infobox may contain cells unrelated to image generation, and thus it may be redundant for image generation.

## Ethical Considerations

In this study, we created our dataset from English Wikipedia. The editors of English

Wikipedia remove unnecessarily offensive content and compile them into an encyclopedia (https://en.wikipedia.org/wiki/Wikipedia:Offensive_material). However, as stated on the official pages (https://en.wikipedia.org/wiki/Wikipedia:Neutral_point_of_view#Bias_in_sources, https://en.wikipedia.org/wiki/Wikipedia:Reliable_sources#Biased_or_opinionated_sources), the current English Wikipedia permits the use of biased information sources. Thus, there is a possibility that our created dataset also inherits the original biases of English Wikipedia.

## Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers JP21K17801, JP23H03458.

## References

Yonatan Bisk, Rowan Zellers, Ronan Le bras, Jianfeng Gao, and Yejin Choi. 2020. Piqa: Reasoning about physical commonsense in natural language. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7432–7439.

Jize Cao, Zhe Gan, Yu Cheng, Licheng Yu, Yen-Chun Chen, and Jingjing Liu. 2020. Behind the scene: Revealing the secrets of pre-trained vision-and-language models. In *Computer Vision – ECCV 2020*, pages 565–580, Cham. Springer International Publishing.

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

Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. 2021. Unifying vision-and-language tasks via text generation. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 1931–1942. PMLR.

Yifan Du, Zikang Liu, Junyi Li, and Wayne Xin Zhao. 2022. A survey of vision-language pre-trained models. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 5436–5443. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

Patrick Esser, Robin Rombach, and Björn Ommer. 2020. Taming transformers for high-resolution image synthesis.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2021. The pile: An 800gb dataset of diverse text for language modeling.

Lovisa Hagström and Richard Johansson. 2022. How to adapt pre-trained vision-and-language models to a text-only input? In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5582–5596, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Taichi Iki and Akiko Aizawa. 2021. Effect of visual extensions on natural language understanding in vision-and-language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2189–2196, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

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

Gen Li, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang. 2020. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):11336–11344.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and

Ilya Sutskever. 2021. Zero-shot text-to-image generation. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8821–8831. PMLR.

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. 2016. Generative adversarial text to image synthesis. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1060–1069, New York, New York, USA. PMLR.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo-Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. 2022. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems*.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. 2016. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2020. Vl-bert: Pre-training of generic visual-linguistic representations. In *International Conference on Learning Representations*.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Sam Bowman, Dipanjan Das, and Ellie Pavlick. 2019. What do you learn from context? probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations*.

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

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. 2022. OFA: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 23318–23340. PMLR.

Chenfei Wu, Jian Liang, Lei Ji, Fan Yang, Yuejian Fang, Daxin Jiang, and Nan Duan. 2022a. Nüwa: Visual synthesis pre-training for neural visual world creation. In *Computer Vision – ECCV 2022*, pages 720–736, Cham. Springer Nature Switzerland.

Xueqing Wu, Jiacheng Zhang, and Hang Li. 2022b. Text-to-table: A new way of information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2518–2533, Dublin, Ireland. Association for Computational Linguistics.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2048–2057, Lille, France. PMLR.

Tian Yun, Chen Sun, and Ellie Pavlick. 2021. Does vision-and-language pretraining improve lexical grounding? In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4357–4366, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Han Zhang, Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. 2021. Cross-modal contrastive learning for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 833–842.

## A Details of the datasets for pre-training OFA

OFA pre-training uses various datasets for pre-training tasks in language, vision, and vision & language modalities, as shown in Table 5. Note that 1.53% of Pile (Gao et al., 2021) listed in Table 5 contains information from English Wikipedia. Therefore, we can understand that although OFA's pre-training focuses on V&L tasks, it is also designed to prevent the knowledge acquired from natural language data from forgetting.

## B Details of the metric calculation

### B.1 Table-$F_1$

Let $e$ be an element of a target cell type. Here, we define a function $Match_{r,g}(e)$ that calculates the exact match of elements in reference and generated tables as follows:

$$\begin{aligned} &Match_{r,g}(e) \\ =&Min(Count_r(e), Count_g(e)), \end{aligned} \tag{1}$$

where $Count_r(e)$, $Count_g(e)$ are functions that return frequencies of $e$ in a generated table $g$ and a reference table $r$, respectively. Note that $Min$ is a function that returns the minimum value from the given one. By using $Match_{r,g}(e)$, we calculate $Table\text{-}F_1$ as follows:

$$P(g,r) = \frac{\sum_{e \in g} Match_{r,g}(e)}{\sum_{e' \in g} Count_g(e')}, \tag{2}$$

$$R(g,r) = \frac{\sum_{e \in r} Match_{r,g}(e)}{\sum_{e' \in g} Count_r(e')}, \tag{3}$$

$$F_1(g,r) = \frac{2P(g,r)R(g,r)}{P(g,r)+R(g,r)}, \tag{4}$$

$$Table\text{-}F_1 = \frac{1}{|D|} \sum_{(g,r) \in (G,R)} F_1(g,r), \tag{5}$$

where $|D|$ denotes a number of tables, $G$ denotes all generated tables, and $R$ denotes all reference tables.

### B.2 Corpus-$F_1$

Instead of $Match_{r,g}(e)$, we define $Match_{R,G}(e)$ as follows:

$$\begin{aligned} &Match_{R,G}(e) \\ =&Min(Count_R(e), Count_G(e)), \end{aligned} \tag{6}$$

where $Count_R(e)$, $Count_R(e)$ are functions that return frequencies of $e$ in all generated tables $G$ and all reference tables $R$, respectively. By using $Match_{R,G}(e)$, we calculate $Corpus\text{-}F_1$ as follows:

$$P(G,R) = \frac{\sum_{e \in G} Match_{R,G}(e)}{\sum_{e' \in G} Count_G(e')}, \tag{7}$$

$$R(G,R) = \frac{\sum_{e \in R} Match_{R,G}(e)}{\sum_{e' \in R} Count_R(e')}, \tag{8}$$

$$Corpus\text{-}F_1 = \frac{2P(G,R)R(G,R)}{P(G,R)+R(G,R)}. \tag{9}$$

## C Groups/Headers/Values in an infobox

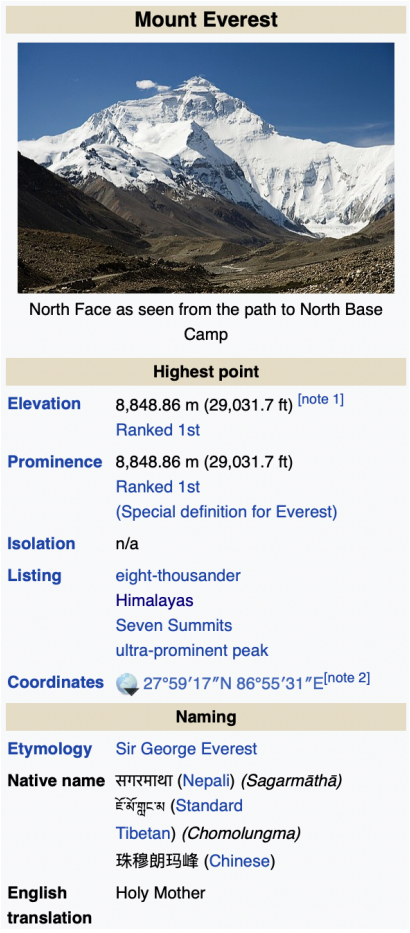



Figure 4: An example infobox with groups[14].

Figure 4 shows an example infobox that includes multiple groups. In this example, we can see two groups named with "Highest point" and "Naming". The headers "Elevation", "Prominence", "Isolation", "Listing", and "Cordinates" are grouped into "Highest point". The headers "Etymology", "Native name", and "English translation" are grouped into "Naming". The headers have corresponding values

[14] https://en.wikipedia.org/wiki/Mount_Everest

| Modality | Task | Dataset |
|---|---|---|
| Vision & Language | Image Captioning<br>Image-Text Matching | CC12M, CC3M, SBU, COCO, VG-Cap |
| | Visual Question Answering | VQAv2, VG-QA, GQA |
| | Visual Grounding<br>Grounded Captioning | RefCOCO, RefCOCO+, RefCOCOg, VG-Cap |
| Vision | Detection<br>Image Infilling | OpenImages, Object365, VG, COCO<br>OpenImages, YFCC100M, ImageNet-21K |
| Language | Masked Language Modeling | Pile |

Table 5: Datasets used for pre-training OFA.

| Type Frequency | | | | |
|---|---|---|---|---|
| **Type** | **Total** | **Train** | **Valid** | **Test** |
| Header | 12,804 | 12,071 | 3,373 | 3,401 |
| Group | 201,937 | 183,728 | 13,252 | 13,444 |
| Value | 772,392 | 705,556 | 54,292 | 55,162 |
| **Appearance Frequency** | | | | |
| **Type** | **Total** | **Train** | **Valid** | **Test** |
| Header | 1,535,791 | 1,383,138 | 75,870 | 76,783 |
| Group | 518,125 | 466,337 | 25,745 | 26,043 |
| Value | 1,535,791 | 1,383,138 | 75,870 | 76,783 |

Table 6: Frequencies for each type of cells in each data split.

such as the value "Holy Mother" to the header "English translation". In the evaluation, we treat values as pairs with including their corresponding headers like ("English translation", "Holy Mother") for the last row of the infobox in Figure 4.

## D Details of our created dataset

Wikipedia HTML dump data contains Wikipedia articles in HTML format, so we extracted infoboxes by using BeautifulSoup[15]. Since the infoboxes contain links to the references of the main article in the form of [#number], we removed them. We filtered out table rows that have more than two columns.

In table generation, if the short side of the input image exceeded 480px, we reduced the short side to 480px while maintaining the aspect ratio. In image generation, we changed the short side of the original image to 256px while maintaining the aspect ratio and then cropped the center of the image with a 256px square.

To measure the performance of both small and large models in the future, we also created additional datasets for the table generation with the short side of the image up to 256px and 384px, respectively. Similarly, we also created a dataset for image generation with both sides of the image set to 128px.

For the sake of future expansion and to avoid data confusion, we divided the collected data into test data if the remainder of the SHA256 value of the title divided by 20 is 0, development data if the remainder is 1, and training data otherwise. Please see Table 2 for the size of the dataset.

Table 6 shows the frequencies of each type of cells used for $F_1$ in §3.1. This result indicates

[15] https://www.crummy.com/software/BeautifulSoup/bs4/doc/

| **Type frequencies of values for each header** | | | | |
|---|---|---|---|---|
| **Split** | **Mean** | **Std.** | **Max** | **Min** |
| All | 60.3 | 548.4 | 18,518 | 1 |
| Train | 58.5 | 516.5 | 17,050 | 1 |
| Valid | 16.1 | 79.1 | 1,506 | 1 |
| Test | 16.2 | 80.3 | 1,557 | 1 |
| **Appearance frequencies of values for each header** | | | | |
| **Split** | **Mean** | **Std.** | **Max** | **Min** |
| All | 119.9 | 1244.0 | 48,150 | 1 |
| Train | 114.6 | 1153.2 | 43,350 | 1 |
| Valid | 22.5 | 118.7 | 2,391 | 1 |
| Test | 22.6 | 119.4 | 2,409 | 1 |

Table 7: Statitics of frequencies for values in each header. Std. denotes standard deviation, Max and Min denote maximum and minimum frequencies, respectively.

| **Split** | **Mean** | **Std.** | **Max** | **Min** |
|---|---|---|---|---|
| All | 17.6 | 7.6 | 149 | 1 |
| Train | 17.6 | 7.6 | 149 | 1 |
| Valid | 17.6 | 7.6 | 99 | 1 |
| Test | 17.5 | 7.6 | 68 | 1 |

Table 8: Statistics for the number of cells in tables. The notations are the same as Table 7.

| Title | Image | BART | OFA (Title & Image) | Reference |
|---|---|---|---|---|
| Low Pike |  | Elevation \| 1,859 m (3,927 ft) <> Location \| England <> Range \| Lake District <> Prominence \| c. 1 m <> Parent peak \| Low Pike <> Topo map \| OS Landranger 89, 90, Explorer OL4 <> OS grid reference \| NN93722 <> Listing \| Marilyn, Hewitt, Nuttall | Elevation \| 1,000 m (1,000 ft) <> Location \| South England <> Coordinates \| 45°49'0"N, 7°10'4"W <> Range \| South East England <> > Range \| south east england <> Topo map \| CDT | Elevation \| 508 m (1,667 ft) <> Range \| Lake District, Eastern Fells <> Prominence \| 28 m <> Parent peak \| Dove Crag <> Topo map \| OS Landranger 90 OS Explorer 7 <> OS grid reference \| NY373077 <> Listing \| Wainwright |
| Ferruginous Pygmy-owl |  | Conservation status <> Least Concern <> Scientific classification <> Kingdom: Animalia Phylum: Chordata Class: Aves Order: Passeriformes Family: Emberizidae Genus: Emberiza Species: E. ferruginus <> Domain: \| Animalia | Conservation status <> Least Concern <> Scientific classification <> Kingdom: Animalia Phylum: Chordata Class: Aves Order: Passeriformes Family: Pterodactylidae Genus: Ferruginous Species: F. cinereus <> kingdom: \| Animalia | Conservation status <> Least Concern <> Scientific classification <> Kingdom: Animalia Phylum: Chordata Class: Aves Order: Strigiformes Family: Strigidae Genus: Glaucidium Species: G. brasilianum <> Kingdom: \| Animalia <> Phylum: \| Chordata <> Class: \| Aves <> Order: \| Strigiformes <> Family: \| Strigidae <> Genus: \| Glaucidium <> Species: \| G. brasilianum <> Binomial name <> Glaucidium brasilianum (Gmelin, 1788) |
| Achlys (plant) |  | Scientific classification <> Kingdom: Plantae Division: Magnoliophyta Class: Liliopsida Order: Asparagales Family: Orchidaceae Subfamily: Higher Epidendroideae Genus: Achlys L. | Scientific classification <> Kingdom: Plantae Division: Magnoliophyta Class: Liliopsida Order: Asterales Family: Asteraceae Genus: Achlys Species: C. lilius <> kingdom: \| Plantae <> Division: \| Magnoliopsida <> Class: \| Liliaceae <> Order: \| Astrales <> Family: \| Asteraceae | Scientific classification <> Kingdom: Plantae Division: Magnoliophyta Class: Magnoliopsida Order: Ranunculales Family: Berberidaceae Genus: Achlys DC. <> Kingdom: \| Plantae <> Division: \| Magnoliophyta <> Class: \| Magnoliopsida <> Order: \| Ranunculales <> Family: \| Berberidaceae <> Genus: \| Achlys DC. <> Species <> 2 or 3 - see text |
| Giant's Castle |  | Developer(s) \| Capcom <> Publisher(s): \| Capcom (Japan) <> Platform(s: \| PlayStation 2 <> Release date \| JP November 15, 2002 NA November 20, 2002 <> Genre(s)/ \| Adventure game <> Mode(es) \| Single player, multiplayer <> Media \| DVD-ROM <> Input methods \| DualShock 2 Giant's Castle | Elevation \| 1,922 metres (1,923 ft) <> Location \| New York, United States <> Coordinates \| 41°44'00"N, 73°48'50"W <> Range \| North York, New York <> Prominence \| 2,944 metres (2,924 ft) | Elevation \| 3,315 metres (10,877 feet) <> Location \| KwaZulu-Natal, South Africa <> Range \| Drakensberg <> Coordinates \| 29°20'S, 29°29'E <> Easiest route \| scramble |

Table 9: Tables generated by BART and OFA with title and image input and those of references.

that all types of cells have large number of type frequencies.

Table 7 shows the statistics of frequencies for values in each header. Note that in Table 7, we do not take into account groups for the calculation different from the $F_1$ in §3.1. From the table, we can understand that frequencies of values for each header have large variances.

Table 8 shows the statistics for the number of cells for each table. This result indicates that tables in infoboxes have the various number of cells.

Taking into account these results, we can understand that predicting cells based only on a label classification setting is difficult due to the various and diverse characteristics of the infobox tables.

To strictly comply with the license, we will only release text data to the public in the dataset release. For images, we will provide their URLs and preprocessing scripts for reproducing our dataset.

## E Details of experimental settings

For both tasks, we modified the publicly available implementation[16] by the authors of OFA. Since the released OFA uses the number of words after splitting by spaces for determining the maximum token length, we modified the OFA to use subwords to specify the maximum token length in the same way as BART. We set the maximum length for input and output in table and image generation to 1024 subwords. In addition, from the perspective of investigating the characteristics of the model and dataset, we used only maximum likelihood estimation for training and did not perform reinforcement learning. We ran training of each model three times with different seeds 0, 1, and 2.

[16] https://github.com/OFA-Sys/OFA (Apache License 2.0).

### E.1 Table Generation

To avoid an unfair comparison of BART and OFA due to different implementations, we transferred BART's weight parameters[17] to OFA and ran BART on OFA. We used the hyperparameters in the summarization of OFA for generation from titles. We also used the hyperparameters in captioning of OFA for generation from images. For a fair comparison, we used the captioning settings for all inferences. When the input includes titles, we used the prompt *What is the infobox of " {ENTITY_NAME} "?*. When the input only includes images, we used the prompt *What is the infobox of the image?*. We performed the text-only experiments with four RTX 3090s in one day and the image-included experiments with four RTX A6000s in one day.

### E.2 Image Generation

Basically, we inherited the hyperparameters used in OFA, but due to learning time, we set the beam size to 1 when generating images in the development data after each epoch in training. We used beam size 24 for testing, the same as in the original setting. We used the prompt *What is the complete image? Caption: {CAPTION}* to generate images. When using tables, we combined the input with the delimiter <> at the end of the original input. We performed each experiment with four RTX A6000s in two days.

## F Generated examples

### F.1 Tables

Table 9 shows the generated tables in the test data. In the first row regarding "Low Pike", BART generated a table for the mountain, whereas OFA generated a table for a city in the United Kingdom. This result is along with the result of the automatic evaluation that BART's prediction performance of values is better than other methods. However, even BART did not specify the detailed location of the mountain. This result indicates the difficulty of storing large amounts of geographic information in a pre-trained model.

In the second row regarding "Ferruginous Pygmy-owl", BART wrongly recognized it as a bunting ("Emberizidae"), at least a bird, and OFA wrongly recognized it as a pterosaur ("Pterodactylidae"). Thus, this is a case that the forgotten knowledge about the entity was not completed with the image.

In the third row regarding "Achlys (plant)", both models recognized it as a plant ("Plantae"), and OFA precisely predicted its division as "Magnoliopsida" by the image. However, both models could not predict further details. This result indicates the difficulty of identifying plants with diverse species.

In the fourth row regarding "Giant's Castle", BART wrongly recognized it as a video game by its misleading name, even though OFA at least recognized it as a building in New York. The result is a case that the image supports the table generation by completing the knowledge about the entity. However, this support is not enough to generate precise information.

### F.2 Images

Table 10 shows the generated images in the test data. In the first row, regarding "Upper Lake (Bhopal)", we can see both settings generated images along with the caption. Since such landscape photographs do not require the depiction of details, it is clear that images can be generated without detailed knowledge.

In the second row regarding "May Lake", only w/ Tab. generated a lake with the mountain corresponding to the information in the table that shows the lake is at a high place. This result indicates that the table information can support generating images based on correct knowledge.

In the third row regarding "Littoral Rockthrush", we can see that both w/ Tab. and w/o Tab. struggled to generate bird images. However, even in this difficult situation, w/ Tab. generated a more precise image than w/o Tab. by using the table information. This result is along with our automatic evaluation results that table information can improve image generation performances.

In the fourth row regarding "Gießen (region)", we can understand from this result that using a table alone is insufficient to generate precise images of geographic information.

We can see interesting results in the fifth row regarding "Giant's Castle", which is a mountain. Both w/o Tab. and w/ Tab. wrongly generated large castles due to the misleading name "Giant's Castle". Furthermore, w/ Tab. generated a large castle that looks like a mountain based on the knowledge of 3,315 meters in the table. This result indicates a limit to disambiguation based solely on the table.

[17] https://dl.fbaipublicfiles.com/fairseq/models/bart.base.tar.gz (MIT License).

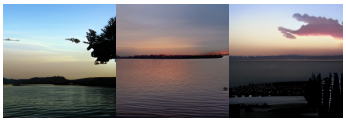
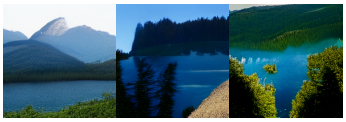
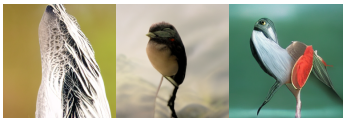
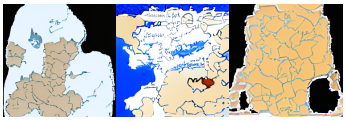
| Input | w/ Tab. | w/o Tab. | Ref. |
|---|---|---|---|
| **Title**: Upper Lake (Bhopal) **Caption**: Upper Lake (Bhopal) - Sunset **Table**: Location \| Madhya Pradesh, Bhopal <> Primary inflows \| Kolans River <> Catchment area \| 361 km² <> Basin countries \| India <> Surface area \| 31 km² |  |  |  |
| **Title**: May Lake **Caption**: May Lake - View from the trail up Mt. Hoffman. **Table**: Location \| Yosemite National Park, California <> Coordinates \| 37°50'50"N, 119°29'37"WCoordinates: 37°50'50"N, 119°29'37"W <> Basin countries \| United States <> Surface elevation \| 9,270 ft (2,830 m) |  |  |  |
| **Title**: Littoral Rock-thrush **Caption**: Littoral Rock-thrush, M. imerinus **Table**: Conservation status <> Least Concern <> Scientific classification <> Kingdom: Animalia Phylum: Chordata Class: Aves Order: Passeriformes Family: Muscicapidae Genus: Monticola Species: M. imerinus <> Kingdom: \| Animalia <> Phylum: \| Chordata <> Class: \| Aves <> Order: \| Passeriformes <> Family: \| Muscicapidae <> Genus: \| Monticola <> Species: \| M. imerinus <> Binomial name <> Monticola imerinus (Hartlaub, 1860, St Augustine Bay, southeast Madagascar) |  |  |  |
| **Title**: Gießen (region) **Caption**: Map of Hesse highlighting the Regierungsbezirk of Gießen **Table**: State \| Hesse <> District seat \| Gießen <> Area \| 5,381.14 km² <> Population \| 1,061,444 (30 Sep. 2005) <> Pop. density \| 197 /km² <> Web page \| www.rp-giessen.de |  |  |  |
| **Title**: Giant's Castle **Caption**: Panorama at Giant's Castle **Table**: Elevation \| 3,315 metres (10,877 feet) <> Location \| KwaZulu-Natal, South Africa <> Range \| Drakensberg <> Coordinates \| 29°20'S, 29°29'E <> Easiest route \| scramble |  |  |  |

Table 10: Generated images. w/ Tab denotes the setting with tables, w/o Tab denotes the setting without tables, and Ref. denotes the reference images.