Effect of Visual Extensions on Natural Language Understanding in Vision-and-Language Models

Taichi Iki[1,2] and **Akiko Aizawa**[1,2]

[1]National Institute of Informatics, Chiyoda-ku, Tokyo, Japan
[2]Graduate University for Advanced Studies, Hayama, Kanagawa, Japan
{iki,aizawa}@nii.ac.jp

Abstract

A method for creating a vision-and-language (V&L) model is to extend a language model through structural modifications and V&L pre-training. Such an extension aims to make a V&L model inherit the capability of natural language understanding (NLU) from the original language model. To see how well this is achieved, we propose to evaluate V&L models using an NLU benchmark (GLUE). We compare five V&L models, including single-stream and dual-stream models, trained with the same pre-training. Dual-stream models, with their higher modality independence achieved by approximately doubling the number of parameters, are expected to preserve the NLU capability better. Our main finding is that the dual-stream scores are not much different than the single-stream scores, contrary to expectation. Further analysis shows that pre-training causes the performance drop in NLU tasks with few exceptions. These results suggest that adopting a single-stream structure and devising the pre-training could be an effective method for improving the maintenance of language knowledge in V&L extensions.

1 Introduction

Pre-trained vision-and-language (V&L) models improve the performance of tasks that require an understanding of the V&L grounding, including visual question answering (Antol et al., 2015), referring expression comprehension (Kazemzadeh et al., 2014), and image-text matching (ITM) (Suhr et al., 2019). Recent V&L tasks, such as multimodal reading comprehension (Kembhavi et al., 2017; Yagcioglu et al., 2018; Hannan et al., 2020; Tanaka et al., 2021) and dialogue (Ilinykh et al., 2019; Haber et al., 2019; Udagawa and Aizawa, 2019), require a deeper NLU as well as the grounding. Extending pre-trained language models (LMs) is an option for those tasks as this allows V&L models to inherit language knowledge from their source LMs. The typical extending consists of visual pre-training and structure such as the stream type; the single-stream inserts vision tokens into the input sequence of the LM, and the dual-stream uses another sequence for early visual encoding.

One of the remaining challenges is to understand how such extensions affect the pre-trained language knowledge. For example, Lu et al. (2019) proposed the dual-stream model where part of the goal was to protect the learned LMs. The authors focused on evaluation with V&L tasks and did not evaluate their models with language-only tasks. Cao et al. (2020) evaluated the extent of language knowledge loss in the single/dual-stream models against the source LM using language-only tasks. However, the difference between single-stream and dual-stream models was unclear because the pre-training was also different in their models.

In this paper, we investigate the effect of visual extensions in V&L models on language-only tasks[1]. Bugliarello et al. (2020) proposed a framework to unify transformer-based V&L models and compared some single/dual-stream models in the same setup. Based on their work, our study shows how these structural differences affect the performance of NLU using the GLUE (Wang et al., 2019) tasks.

In our experiments, fine-tuning of pre-trained V&L models shows that both single/dual-stream models perform worse than the source LM and that single-stream models perform slightly better than dual-stream models. Further, we fine-tune the models created by only structural modifications without pre-training. We observe that the single/dual modification alone has little effect on the GLUE scores, indicating the performance degradation is primarily caused by pre-training. We also see how the V&L models changed from the source LM by analyzing the changes in the model parameters and the problem sets that each model can solve. Our results

[1]The source code for our experiments is available at https://github.com/Alab-NII/eval_vl_glue

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 2189–2196
November 7–11, 2021. ©2021 Association for Computational Linguistics

suggest that it would be more effective to adopt a single stream, and devise pre-training strategies for maintaining language knowledge.

2 Controlled V&L Models

In this section, we describe the pre-trained V&L models used in our experiments. Bugliarello et al. (2020) proposed a framework for V&L models that consider a sequence of tokens in sentences as language information, and a sequence of recognized object regions as visual information. In their framework, they reproduced five existing models, VisualBERT (Li et al., 2019), Uniter (Chen et al., 2020), VL-BERT (Su et al., 2020), ViLBERT (Lu et al., 2019), and LXMERT (Tan and Bansal, 2019), and made their controlled versions by modifying some parts for a fairer and easier comparison. We use these controlled versions.

2.1 Structural Modification

We describe streams and embeddings, which are the basic factors of the model structures. We summarize the model structures in the controlled setup used in this experiment in Table 1.

Streams. V&L models can be divided into two categories based on how the vision and language sequences are encoded. Single-stream models, VisualBERT, Uniter, and VL-BERT, jointly process the vision and language sequences in a single encoder. Dual-stream models, ViLBERT and LXMERT, encode those sequences separately before encoding them jointly. ViLBERT is an early example of the dual-stream models and was proposed mainly to account for the differences in abstraction levels between vision and language, and to protect learned language models. In the controlled setup of Bugliarello et al. (2020), the stream type is identical to the original one in all models.

Embeddings. The major difference in embeddings is the use of global visual feature. The original VisualBERT, Uniter, and LXMERT do not use the global visual feature. ViLBERT has a token that represents the global visual feature at the beginning of vision sequences. VL-BERT inserts the global visual feature to the last of vision sequences and also adds the global visual feature to each token embedding in the language sequence. Object location is also expressed differently. The original VL-BERT and LXMERT use four attributes (left, top, right, bottom). In addition to the four attributes, the original ViLBERT uses area, and the original Uniter uses width, height, and area. VisualBERT does not use location information[2].

The controlled setup is based on the structure of ViLBERT. For the global image feature, the setup inserts the average of vision tokens to the head of the vision sequence for all models. In addition to inserting the global visual feature, the controlled VL-BERT adds it to the respective tokens in the language sequence. For location, VisualBERT's setup that do not use location information remain the same, while the other models use the five attributes. The five attributes are normalized by width or height. Another point is the token type for the vision tokens. In the controlled setup, the token type is not added for ViLBERT and LXMERT because they have separate streams. Of the single-stream models, VisualBERT and Uniter use BERT's token type ID to specify vision tokens, while VL-BERT adds a new embedding to represent vision tokens.

2.2 Pre-training

We summarize the pre-training used in the controlled setup to train the five model structures described above. Note that we omit the detail of the pre-training used in each original paper here.

The five models were pre-trained on Google's Conceptual Captions (Sharma et al., 2018) corpus, which was collected from Web images and their alt-text HTML attributes. The corpus was filtered before training, and the size was approximately 2.7 M pairs as a result. Three tasks, masked language modelling (MLM), masked object classification (MOC), and ITM, were made from image-text pairs in the corpus. Given an image-text pair, the model predicts masked language tokens for MLM, the object class of masked vision tokens for MOC, and whether the pair is correct or not for ITM.

The weights of the five models were initialized with the pre-trained weights of $\text{BERT}_{\text{BASE}}$ if the corresponding weights were in $\text{BERT}_{\text{BASE}}$; otherwise (e.g., the weights of the vision encoder in dual-stream models), they were initialized randomly.

3 Experiment with GLUE

3.1 Datasets

The GLUE benchmark (Wang et al., 2019) is a collection of diverse tasks for studying NLU systems.

[2]If alignments between words and regions are provided, VisualBERT adds the same position embeddings to matched word and region tokens instead.

Structure	Abbreviation in this paper	Stream	#param	Location format	Global image feat.	Vision type ID	Original paper
VisualBERT$_{\text{CTRL}}$	VIS$_{\text{CTRL}}$	Single	112M	not used	head	from BERT	Li et al. (2019)
Uniter$_{\text{CTRL}}$	UNI$_{\text{CTRL}}$		112M	LTRBA	head	from BERT	Chen et al. (2020)
VL-BERT$_{\text{CTRL}}$	VL$_{\text{CTRL}}$		114M	LTRBA	head + added to each word	extended	Su et al. (2020)
ViLBERT$_{\text{CTRL}}$	VIL$_{\text{CTRL}}$	Dual	240M	LTRBA	head	not used	Lu et al. (2019)
LXMERT$_{\text{CTRL}}$	LX$_{\text{CTRL}}$		209M	LTRBA	head	not used	Tan and Bansal (2019)

Table 1: Comparison of the structures in the **controlled** setup used in this study. L, T, R, B, and A in the location format column denote left, top, right, bottom, and area, respectively. Bugliarello et al. (2020)'s controlled setup unifies the use of the location format and global visual features, which were different in the original proposals.

It consists of nine tasks: CoLA (Warstadt et al., 2019), SST-2 (Socher et al., 2013), MRPC (Dolan and Brockett, 2005), QQP[3], STS-B (Cer et al., 2017), MNLI (Nangia et al., 2017), QNLI (Rajpurkar et al., 2016), RTE (Dagan et al., 2005; Bar Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009), and WNLI (Levesque et al., 2012). STS-B is a single-valued regression task, and the others are classification tasks. We train the controlled pre-trained models on the training sets and evaluate them with the development sets. Figure 1 (left) shows the number of the training sentences in the corpora and their word overlap between the corpus used in the V&L pre-training.

3.2 Implementation Details

We fine-tuned pre-trained models published by Bugliarello et al. (2020) [4]. To use a script for the GLUE benchmark, we modified the model codes for Huggingface's Transformers (Wolf et al., 2020)[5]. We used the BERT-uncased tokenizer to tokenize sentences.

Image inputs. Because the GLUE tasks have no image input, we used a black image (of 224×224 pixels) in our experiments. We followed the method of Bugliarello et al. (2020) for image processing; we input the images to the Faster R-CNN object detector (Ren et al., 2015) trained for the Bottom-Up and Top-Down model (Anderson et al., 2018), and used the top 36 detected results (bounding boxes and feature vectors) as vision tokens[6]. We used the average of the vision tokens as a global visual token. Those vision tokens were fixed and used for both training and evaluation in all models.

In this study, we tried image completion with black images for tasks where no image is provided as a simple way to preserve the input format used in pre-training. However, there are many possible methods for complementing the image input. For example, a method as simple as the present one can use other images, a noise input, or learnable parameters. Examining the impact of image input completion methods remains as future work.

Head for classification. We adopted the method used in Bugliarello et al. (2020) for V&L tasks. We used a learnable linear layer to calculate the likelihood of document classes, such as entailment/neutral/contradiction. We input the element-wise product of two vectors made from the model's output sequence into the linear layer. For those two vectors, we pooled the portions of the model's output sequence that correspond to the vision input and to the language input, respectively, by taking the first token of each portion. This corresponds to taking the outputs of the [CLS] token (in the language sequence) and the global visual token.

Hyperparameters for fine-tuning. We used a batch size of 64 and Adam for optimization. The learning rate was initialized at 2e-5 and decreased linearly. We trained for five epochs, evaluating the loss on the dev sets at the end of each epoch. Finally, we adopted the model with the lowest loss.

3.3 Overall Result

Table 2 shows the results of the GLUE benchmark. In our experiment, we fine-tuned five V&L models and their source language model–BERT$_{\text{BASE}}$. We also cited the BiLSTM baseline from the GLUE paper. The Glue avg of five V&L models decrease compared to BERT$_{\text{BASE}}$. We can see a trend where the single-stream models perform slightly better than the dual-stream models. Note that this trend is consistent with the results of Cao et al. (2020) for linguistic probing of the original Uniter and LXMERT. Although the difference is small, this suggests that the single-stream models can main-

[3] https://www.kaggle.com/c/quora-question-pairs

[4] https://github.com/e-bug/volta/blob/main/MODELS.md

[5] We checked that our implementation reproduced original results with a V&L task, NLVR^2 (Suhr et al., 2019).

[6] Although the image was monochromatic black, 36 bounding boxes with different features were detected.

	GLUE (Language)										V&L
	avg↑ (SD)	**CoLA**	**SST-2**	**MRPC**	**QQP**	**STS-B**	**MNLI**	**QNLI**	**RTE**	**WNLI**	**avg↑**
BiLSTM	66.7	17.6	87.5	77.9/85.1	85.3/82.0	71.6/72.0	66.7	77.0	58.5	56.3	
$\text{BERT}_{\text{BASE}}$	77.3 (0.8)	54.6	92.5	81.9/87.6	90.6/87.4	88.2/87.9	84.4	91.0	62.5	48.8	
Model avg	71.6	38.0	88.9	70.1/80.8	89.0/85.5	78.5/78.7	81.0	85.5	55.7	52.6	68.0
VIS_{CTRL}	**72.5** (1.2)	38.6	89.4	**71.9/82.1**	**89.4/86.0**	81.8/81.7	**81.8**	**87.0**	56.6	53.1	69.2
UNI_{CTRL}	71.4 (0.3)	37.4	89.7	69.3/80.3	89.2/85.7	74.9/75.6	81.2	86.0	55.6	**55.4**	**69.7**
VL_{CTRL}	**72.4** (0.8)	38.7	89.8	70.6/81.8	89.0/85.4	**82.9/82.8**	81.4	86.3	55.7	53.1	67.7
VIL_{CTRL}	70.9 (0.8)	36.1	**90.4**	69.0/79.4	88.6/85.0	77.7/78.0	80.1	83.8	53.7	**55.4**	**69.8**
LX_{CTRL}	70.5 (0.2)	**39.0**	90.2	69.8/80.4	89.0/85.4	75.3/75.3	80.7	84.2	**57.2**	46.0	63.6

Table 2: Performance of the development sets of the GLUE tasks (single-task training). The best scores among the five V&L models are shown in bold. We report the Matthews correlation for CoLA; accuracy/F1 for MRPC and QQP; the Pearson/Spearman correlation for STS-B; and the accuracy for all other tasks. For MNLI, we show accuracy averaged over the matched and mismatched sets. The values of BiLSTM are cited from Wang et al. (2019). The other values related to GLUE are our results. We fine-tuned the pre-trained models for each task three times with different random seeds. We show the standard deviation in parentheses for avg and in Appendix B for each task. In the last column, we also show the scores of V&L tasks calculated by averaging the results in Bugliarello et al. (2020). The detail is described in Section 4.3.

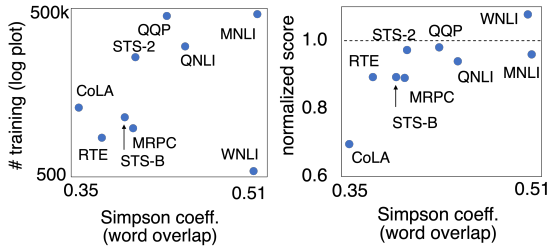

Figure 1: Left: The number of training sentences vs. the Simpson coefficient between the GLUE and CC (training) corpora. Right: The correlation between the Simpson coefficient and the model score. The model scores were averaged over the five V&L models and normalized with $\text{BERT}_{\text{BASE}}$'s score.

tain more of $\text{BERT}_{\text{BASE}}$'s knowledge.

Performance of each task. V&L models perform lower than the BiLSTM baseline for some tasks, including MRPC, RTE, and WNLI. Figure 1 (right) shows the correlation between the word overlap between the corpus for pre-training and the GLUE task corpora and the GLUE score. We can see a positive correlation between those two variables. Although we do not conclude clearly because word overlap and the number of training data also correlate, word overlap could have a large impact on task performance.

4 Analysis

4.1 Amount of Change in Parameters

We expected the model inference to be closer to BERT's inference if a model has parameters closer to BERT. Therefore, we calculated the cosine similarity of the corresponding parameters between pre-trained models and BERT to indicate the degree to which the parameters had changed. Table 3 shows the averaged cosine similarity. We flattened parameters and calculated their similarity as vectors. We can see that the parameters of the single-stream and dual-stream models changed by the same extent. This suggests that separating streams alone may not be sufficient for knowledge maintenance.

	weight	**weight (LN)**	**bias**	**bias (LN)**
#layers	75	25	72	25
VIS_{CTRL}	0.9218	0.9999	0.9963	0.9973
UNI_{CTRL}	0.9197	0.9999	0.9966	0.9971
VL_{CTRL}	0.9193	0.9999	0.9964	0.9968
VIL_{CTRL}	0.9218	0.9999	0.9934	0.9895
LX_{CTRL}	0.9208	0.9998	0.9926	0.9935

Table 3: Averaged cosine similarity between the corresponding parameters in the $\text{BERT}_{\text{BASE}}$ and V&L models. #layers represents the number of layers transferred from $\text{BERT}_{\text{BASE}}$ to V&L models. We computed the averaged similarity of the weights and biases in the layer normalization (LN) layers and the other layers.

	Successful models			
	Both	**$\text{BERT}_{\text{BASE}}$**	**V&L**	**Neither**
VIS_{CTRL}	0.722	0.080	0.049	0.150
UNI_{CTRL}	0.717	0.085	0.050	0.149
VL_{CTRL}	0.700	0.102	0.053	0.146
VIL_{CTRL}	0.710	0.091	0.049	0.150
LX_{CTRL}	0.691	0.111	0.065	0.134

Table 4: Analysis of which models were successful in answering the classification task. STS-B was excluded because it is a regression task. We defined success in a problem as answering correctly in at least two out of three runs.

4.2 Breakdown of Classification Results

Table 4 shows the results of aggregating the GLUE classification task problems into four categories: solvable by both $\text{BERT}_{\text{BASE}}$ and V&L models, $\text{BERT}_{\text{BASE}}$ only, V&L model only, and neither model. We defined success in a given problem as answering correctly in at least two out of three experimental runs. To make Table 4, we first cal-

	Mod. only	Mod.+V&L PT
VIS_{CTRL}	77.4 (1.00)	**72.5 (0.94)**
UNI_{CTRL}	77.9 (1.01)	71.4 (0.92)
VL_{CTRL}	39.5 (0.51)	**72.4 (0.94)**
VIL_{CTRL}	75.6 (0.98)	70.9 (0.92)
LX_{CTRL}	**78.4 (1.01)**	70.5 (0.91)

Table 5: Effect of V&L pre-training on the averaged GLUE score. Values in parentheses are scores that have been normalized by the $\text{BERT}_{\text{BASE}}$ scores.

culated the tables of successful models for each GLUE task and V&L model and second averaged the tables for the tasks. For all five models, there are approximately 5% of problems that they only can solve. This category shows the positive impact of V&L pre-training on NLU. Problems that both models can solve tended to be more common for the single-stream models. This supports the finding that these models retain more language knowledge.

The difference of corpora for the last pre-training between BERT (mainly English Wikipedia) and the V&L models (images' alt-texts) might affect the complexity of the sentences in the problem sets that can be solved only by BERT and only by the V&L models. Thus, we analyzed the distributions of some metrics (sentence length, readability). However, we found no significant difference between the two sets in each model. We show the distributions in Appendix C.

4.3 Language and V&L Tasks

The last column of Table 2 shows the V&L scores for the V&L models. We calculated these scores by averaging the results on the five V&L tasks reported in Bugliarello et al. (2020). Their tasks cover four groups widely used to test V&L models: VQA, image–text retrieval, referring expressions, and multi-modal verification. Comparing the V&L and GLUE scores, we cans see that no model is best in both respects at the same time. There is room for improvement in the V&L extension.

4.4 Structural Modification or Pre-training: Which Has the Greater Impact?

To further analyze the impact of structural modification, we fine-tuned models with only structural modifications (Mod. only). Table 5 shows a comparison between the GLUE scores of the Mod-only models and the full models (Mod+V&L-PT). Except for VL_{CTRL}, the Mod-only models achieve a score comparable to $\text{BERT}_{\text{BASE}}$, and the GLUE score decreases for the Mod+V&L-PT models. The fact that the structural modification preserves the score of the GLUE tasks in most cases suggests that the main factor for the drop in the GLUE tasks is V&L pre-training. This observation emphasizes the impact of pre-training on maintaining the language knowledge. Note that a possible reason for the exception of VL_{CTRL} is that the global visual feature added to the language embeddings may break the language knowledge.

5 Discussion and Conclusion

The number of V&L model works that focus on both V&L tasks and language-only tasks has increased (Ororbia et al., 2019; Lin et al., 2021; Li et al., 2020; Hu and Singh, 2021). Ororbia et al. (2019) proposed a V&L neural architecture and trained it on a language model in a visual context. They demonstrated that their architecture outperforms its equivalent trained on language alone in perplexity and stated that language is inseparable from its physical context. Although it is not clear whether methods that improve the perplexity of language modeling can also apply to maintain the performance of downstream tasks, the strategy of improving models with reference to human cognition would be an important direction. More recently, Li et al. (2020) achieved better performance on language-only tasks than their base model with pre-training on three types of corpora (text, image, and image–text pairs) at the same time. Lin et al. (2021) reported that adding separated extractors for vision and language on top of a single-stream encoder can help maintain language knowledge.

In this paper, we fine-tuned V&L models extended from a language model (LM) to an NLU benchmark to compare their NLU performance. We used five V&L models, including single-stream and dual-stream models, pre-trained in the same setup. The benchmark scores of those models decreased compared with their source LM. We also found that the single-stream models tended to retain (slightly) more language knowledge than the dual-stream models, and that the main cause of the drop in the NLU tasks can be pre-training. Our observations suggest that adopting a single stream and devising pre-training strategies could be effective, at least for preserving the language knowledge.

Acknowledgements

We would like to thank Takuma Udagawa, Taku Sakamoto, and the anonymous reviewers for their insightful comments. This work was supported by JSPS KAKENHI Grant Number 21H03502.

References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual Question Answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433.

Roy Bar Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second PASCAL recognising textual entailment challenge.

Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. 2009. The Fifth PASCAL Recognizing Textual Entailment Challenge. In *Proceedings of the Text Analysis Conference (TAC'09)*.

Emanuele Bugliarello, Ryan Cotterell, Naoaki Okazaki, and Desmond Elliott. 2020. Multimodal Pretraining Unmasked: Unifying the Vision and Language BERTs. *arXiv preprint:2011.15124*.

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 *The 2020 European Conference on Computer Vision*, pages 565–580.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation*, pages 1–14.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. UNITER: Learning UNiversal image-TExt representations. In *The 2020 European Conference on Computer Vision*, pages 104–120.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The PASCAL Recognising Textual Entailment Challenge. In *Proceedings of the First International Conference on Machine Learning Challenges: Evaluating Predictive Uncertainty Visual Object Classification, and Recognizing Textual Entailment*, page 177–190.

William B. Dolan and Chris Brockett. 2005. Automatically Constructing a Corpus of Sentential Paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing*.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The Third PASCAL Recognizing Textual Entailment Challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9.

Janosch Haber, Tim Baumgärtner, Ece Takmaz, Lieke Gelderloos, Elia Bruni, and Raquel Fernández. 2019. The PhotoBook Dataset: Building Common Ground through Visually-Grounded Dialogue. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1895–1910.

Darryl Hannan, Akshay Jain, and Mohit Bansal. 2020. ManyModalQA: Modality Disambiguation and QA over Diverse Inputs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7879–7886.

Ronghang Hu and Amanpreet Singh. 2021. Transformer is All You Need: Multimodal Multitask Learning with a Unified Transformer. *arXiv preprint arXiv:2003.13198*.

Nikolai Ilinykh, Sina Zarrieß, and David Schlangen. 2019. Meetup! a corpus of joint activity dialogues in a visual environment. *arXiv preprint arXiv:1907.05084*.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. ReferItGame: Referring to Objects in Photographs of Natural Scenes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 787–798.

Aniruddha Kembhavi, Minjoon Seo, Dustin Schwenk, Jonghyun Choi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4999–5007.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. The Winograd Schema Challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, page 552–561.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. VisualBERT: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.

Wei Li, Can Gao, Guocheng Niu, Xinyan Xiao, Hao Liu, Jiachen Liu, Hua Wu, and Haifeng Wang. 2020. UNIMO: Towards Unified-Modal Understanding and Generation via Cross-Modal Contrastive Learning. *arXiv preprint arXiv:2012.15409*.

Junyang Lin, An Yang, Yichang Zhang, Jie Liu, Jingren Zhou, and Hongxia Yang. 2021. M6-v0: Vision-and-Language Interaction for Multi-modal Pretraining. *arXiv preprint arXiv:2003.13198*.

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.

Nikita Nangia, Adina Williams, Angeliki Lazaridou, and Samuel Bowman. 2017. The RepEval 2017 Shared Task: Multi-Genre Natural Language Inference with Sentence Representations. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 1–10.

Alexander Ororbia, Ankur Mali, Matthew Kelly, and David Reitter. 2019. Like a baby: Visually situated neural language acquisition. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5127–5136.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*, volume 28.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642.

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*.

Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. 2019. A Corpus for Reasoning about Natural Language Grounded in Photographs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6418–6428.

Hao Tan and Mohit Bansal. 2019. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 5100–5111.

Ryota Tanaka, Kyosuke Nishida, and Sen Yoshida. 2021. VisualMRC: Machine Reading Comprehension on Document Images. In *the AAAI Conference on Artificial Intelligence*.

Takuma Udagawa and Akiko Aizawa. 2019. A natural language corpus of common grounding under continuous and partially-observable context. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7120–7127.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *International Conference on Learning Representations*.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural Network Acceptability Judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Semih Yagcioglu, Aykut Erdem, Erkut Erdem, and Nazli Ikizler-Cinbis. 2018. RecipeQA: A Challenge Dataset for Multimodal Comprehension of Cooking Recipes. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1358–1368.

A Dataset Statistics

Dataset	Task	Size	#vocab.	Word ov btw CC
V&L pre-training				
CC	CAP	2.8M	48,360	1
CC (val)	CAP	14K	10,442	0.63
GLUE benchmark				
WNLI	NLI	635	1.622	0.08
RTE	NLI	2.5K	23,341	0.24
MRPC	P/S	3.7K	13,926	0.26
STS-B	P/S	5.7K	16,436	0.25
CoLA	SS	8.6K	7,845	0.19
SST-2	SS	67K	14,816	0.26
QNLI	NLI	104K	148,413	0.29
QQP	P/S	364K	193,041	0.28
MNLI	NLI	393K	167,790	0.34

Table 6: Training dataset statistics. CC: The Conceptual Captions dataset (Sharma et al., 2018). CAP: image captioning, P/S: paraphrase/similarity task, SS: single-sentence task.

B Additional Data for Overall Results

Table 7 shows the SDs to the averaged scores of V&L models on the GLUE tasks' development sets.

	avg	CoLA	SST-2
$\text{BERT}_{\text{BASE}}$	77.3 (0.8)	54.6 (1.1)	92.5 (0.1)
VIS_{CTRL}	72.5 (1.2)	38.6 (7.3)	89.4 (0.4)
UNI_{CTRL}	71.4 (0.3)	37.4 (6.5)	89.7 (0.5)
VL_{CTRL}	72.4 (0.8)	38.7 (1.5)	89.8 (0.9)
VIL_{CTRL}	70.9 (0.8)	36.1 (6.0)	90.4 (0.5)
LX_{CTRL}	70.5 (0.2)	39.0 (6.1)	90.2 (0.5)

	MRPC	QQP	STS-B
$\text{BERT}_{\text{BASE}}$	81.9 / 87.6 (0.6) / (0.5)	90.6 / 87.4 (0.0) / (0.1)	88.2 / 87.9 (0.3) / (0.3)
VIS_{CTRL}	71.9 / 82.1 (1.4) / (0.8)	89.4 / 86.0 (0.1) / (0.1)	81.8 / 81.7 (4.0) / (3.6)
UNI_{CTRL}	74.9 / 75.6 (2.0) / (2.2)	69.3 / 80.3 (0.8) / (0.7)	89.2 / 85.7 (0.1) / (0.1)
VL_{CTRL}	70.6 / 81.8 (0.5) / (0.3)	89.0 / 85.4 (0.3) / (0.4)	82.9 / 82.8 (2.3) / (1.9)
VIL_{CTRL}	69.0 / 79.4 (1.3) / (2.1)	88.6 / 85.0 (0.2) / (0.1)	77.7 / 78.0 (1.2) / (0.9)
LX_{CTRL}	69.8 / 80.4 (1.3) / (1.1)	89.0 / 85.4 (0.1) / (0.2)	75.3 / 75.3 (0.8) / (0.7)

	MNLI	QNLI	RTE	WNLI
$\text{BERT}_{\text{BASE}}$	84.2 (0.1)	91.0 (0.4)	62.5 (1.5)	48.8 (5.8)
VIS_{CTRL}	81.6 (0.2)	87.0 (1.1)	56.6 (1.9)	53.1 (4.6)
UNI_{CTRL}	80.9 (0.4)	86.0 (1.0)	55.6 (2.4)	55.4 (1.3)
VL_{CTRL}	81.2 (0.2)	86.3 (0.1)	55.7 (1.4)	53.1 (3.5)
VIL_{CTRL}	79.9 (0.5)	83.8 (0.6)	53.7 (0.9)	55.4 (1.8)
LX_{CTRL}	80.4 (0.2)	84.2 (0.2)	57.2 (3.4)	46.0 (9.2)

Table 7: Standard deviations of our results in the performance on the GLUE tasks' development sets (Table 2). SDs are shown in parentheses below each value. We ran three experiments for each task.

C Additional Data for Analysis

We show the distributions of sentence length and readability mentioned in Section 4.2 in Figure 2 and Figure 3, respectively.

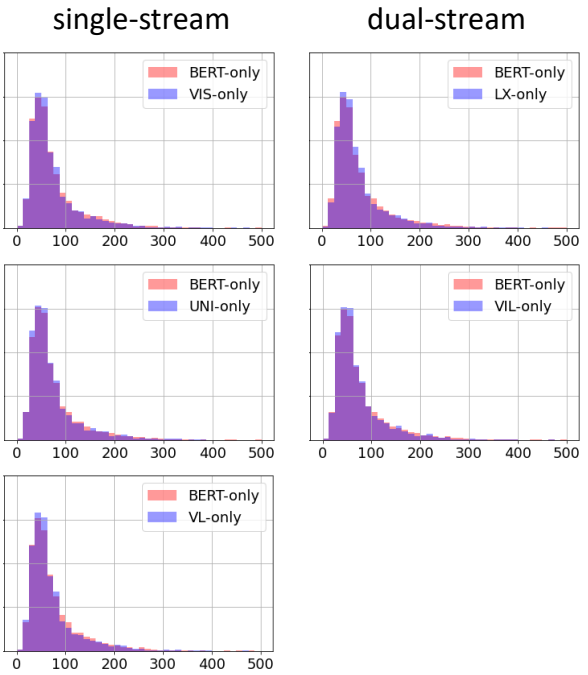

Figure 2: The sentence length distributions in the problem sets solved only by the V&L model and only by BERT. In each plot, the area of the distribution is normalized to 1. The range of the vertical axis is [0, 0.020].

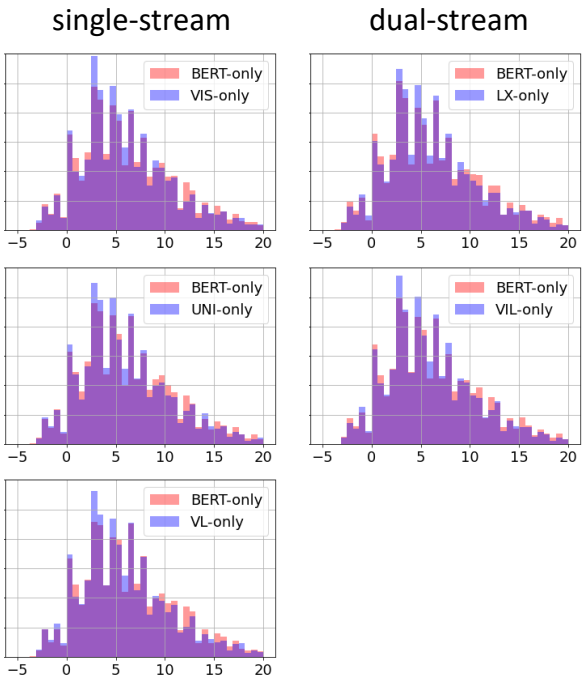

Figure 3: The Flesch–Kincaid Grade Level distributions of sentences in the problem sets solved only by the V&L model and only by BERT. In each plot, the area of the distribution is normalized to 1. The range of the vertical axis is [0, 0.15].