Does Vision-and-Language Pretraining Improve Lexical Grounding?

Tian Yun and **Chen Sun** and **Ellie Pavlick**
Brown University
{tian_yun, chensun, ellie_pavlick}@brown.edu

Abstract

Linguistic representations derived from text alone have been criticized for their lack of *grounding*, i.e., connecting words to their meanings in the physical world. Vision-and-Language (VL) models, trained jointly on text and image or video data, have been offered as a response to such criticisms. However, while VL pretraining has shown success on multimodal tasks such as visual question answering, it is not yet known how the internal linguistic representations themselves compare to their text-only counterparts. This paper compares the semantic representations learned via VL vs. text-only pretraining for two recent VL models using a suite of analyses (clustering, probing, and performance on a commonsense question answering task) in a language-only setting. We find that the multimodal models fail to significantly outperform the text-only variants, suggesting that future work is required if multimodal pretraining is to be pursued as a means of improving NLP in general.

1 Introduction

Large pretrained language models (LMs)–e.g., BERT (Devlin et al., 2019), GPT (Radford et al., 2019; Brown et al., 2020)– derive representations of words and sentences by distilling patterns that exist in large text corpora. While such representations have shown strong empirical performance on many benchmark language understanding tasks, they have been criticized for their lack of *grounding*, i.e., the ability to connect words to the real-world entities, events, and ideas to which they refer. While grounding is obviously necessary for mulitmodal language understanding tasks (e.g., identifying a dog in an image), it has further been argued to be fundamental for learning semantic representations in general. For example, Bender and Koller (2020) argues that models trained without grounding will ultimately fail on some text-only tasks such as goal-oriented dialogue, and Merrill et al. (2021)

argues that an embedding space learned from text alone cannot encode the correct conceptual structure. One proposed solution is to shift from text-only models to multimodal models, which learn to associate language with representations of the non-linguistic world (Bisk et al., 2020a). Such approaches are intuitively appealing, but have not yet been rigorously analyzed in practice.

We test the hypothesis that grounded pretraining yields better linguistic representations (of words and sentences) than does text-only pretraining. For two recently released vision-and-language (VL) models, VideoBERT and VisualBERT, we compare the performance of the multimodal model to a text-only variant. We measure how well the representations encode 1) common sense inferences about the physical world, 2) the semantic structure of verbs and their arguments, and 3) compositional information about objects and their properties. Overall, we do not find evidence that the linguistic representations learned via multimodal pretraining differ meaningfully from those learned from text alone. We argue that such results do not imply that grounding is unimportant for language understanding, but rather that substantial future work on how to combine modalities is required if multimodal methods are to impact NLP in general. Our code is available at https://github.com/tttyuntian/vlm_lexical_grounding.

2 Related Work

Analyzing Pretrained LMs. There has been substantial prior work on analyzing pretrained LMs and the linguistic properties of their representations, looking, e.g., at syntactic parse structure (Hewitt and Manning, 2019; Linzen et al., 2016), semantic structure such as semantic roles and coreference (Tenney et al., 2019a), lexical semantics (Chronis and Erk, 2020; Vulić et al., 2020), and lexical composition (Yu and Ettinger, 2020). Particularly relevant to our studies is prior work which has explored

4357

how well text-only models capture commonsense knowledge about the physical world via intrinsic (Ettinger, 2020; Forbes et al., 2019) and extrinsic (Zellers et al., 2018, 2019; Bisk et al., 2020b) measures. Despite the interest in representations of the non-linguistic world, such analyses have not, to our knowledge, been run on multimodal LMs.

Vision-and-Language Pretraining. There is a long history of multimodal distributional semantics models (Howell et al., 2005; Lazaridou et al., 2015), to which pretrained transformer-based models are the latest addition (Sun et al., 2019; Li et al., 2020). Evaluations of these recent vision-and-language (VL) models has tended to focus on inherently multimodal tasks , e.g., image and video captioning (Sun et al., 2019), visual question answering (Li et al., 2020), or instruction following in robotics (Majumdar et al., 2020). Cao et al. (2020) describes a series of "probing" analyses for multimodal language representations, but focuses on explicit grounding, e.g., to where do models attend in the image when processing "dog"? Little work has analyzed whether the presence of grounded training data impacts the linguistic representations in general. Work that does perform exploratory analyses of the multimodal conceptual representations (Tan and Bansal, 2020; Radford et al., 2021) does not include analysis of comparable text-only models, limiting the conclusions that can be drawn.

3   Vision-and-Language Pretraining

This section describes pretraining approaches that use both vision and language information. In particular, we focus on two that extend the BERT (Devlin et al., 2019) pretraining for text, VideoBERT (Sun et al., 2019) and VisualBERT (Li et al., 2020). Both are single-stream models which directly combine visual and text information at the model inputs, and are trained on paired video+speech and image+caption data, respectively.

More specifically, VideoBERT encodes video data by vector quantization, mapping visual features extracted from 1.5 seconds long video segments into "visual words" with K-Means clustering. The authors downloaded around 300K publicly available cooking videos from YouTube, and obtained the human speech data from YouTube's automatic speech recognition system. Sequences of visual words and speech that are temporally aligned in the original videos are concatenated and fed into a $BERT_{base}$ encoder. Similarly, VisualBERT con-

catenates image region embeddings derived from pretrained object detectors, with their corresponding image captions. The model is pretrained on the COCO (Chen et al., 2015) dataset which contains images and five human annotated captions per image. Both pretraining methods rely on the BERT pre-training objectives, modified to their multimodal setups. Specifically, the objectives contain two parts: (1) a masked language modeling (MLM) objective to predict masked out tokens (VideoBERT predicts both visual and text tokens, while VisualBERT predicts only text tokens) and (2) a visual-language prediction objective, which predicts whether the visual and language sequences come from the same video/image or not.

VL pretraining setup. For VideoBERT, we obtained the training data and pretrained checkpoints from the authors. For VisualBERT, we downloaded the pretrained VisualBERT-NLVR checkpoint [1] pretrained on the Karpathy train split (Karpathy and Fei-Fei, 2015) of COCO (Chen et al., 2015). We refer to these two multimodal pretrained checkpoints as $VideoBERT_{VL}$ and $VisualBERT_{VL}$. Both $VideoBERT_{VL}$ and $VisualBERT_{VL}$ are based on the $BERT_{base}$ architecture, with the difference that our obtained $VideoBERT_{VL}$ was trained from scratch (to ensure a controlled comparison to the text-only model, see below), while the public $VisualBERT_{VL}$ is initialized with its text-only counterpart.

Text-only pretraining setup. For comparison, we train text-only counterpart for each model, $VideoBERT_{text}$ and $VisualBERT_{text}$, using the same text data as the VL model (i.e., the transcribed speech, the captions), while the image data is removed (i.e. the "visual tokens" of a video or an image). Text-only models are pretrained with masked language modeling objective and next sentence prediction objective, since there are multiple sentences of descriptions of a video (VideoBERT) and multiple captions of an image (VisualBERT). We follow the multimodal pretraining setups as faithfully as possible: we used the same $BERT_{base}$ encoder with their corresponding initialization method, the same maximum sequence length, as well as other optimization hyperparameters such as learning rate and number of training epochs. Therefore, the VL models and text-only counterparts have the same architecture and the same number of parameters: VideoBERT models have 125M parameters, while

---

[1] https://github.com/uclanlp/visualbert.

VisualBert models have 109M parameters. More details can be found in Appendix A.2.

Limitations. Our experiments are based on two popular variants of VL pretraining frameworks. We picked these two models as they reflect the common trends in VL pretraining for videos and images, and their model architectures and pretraining objectives closely resemble the BERT model, making it easier to compare with their text-only counterparts. However, this comes with the limitation that the models we analyze are trained on data of a different domain than many of our evaluation tasks (e.g., the data for VideoBERT comes from cooking videos on YouTube while the probing tasks are drawn largely from general web text). Thus, absolute results must be interpreted with this domain mismatch in mind. That said, our inclusion of a text-only baseline still allows us to isolate the benefit of the visual modality in an apples-to-apples comparison. Ideally, we would train VL models on multimodal corpora which match the evaluation domains. However, such corpora simply do not exist at the time of writing. Thus, despite the limitations due to domain, our results are representative of the current benefits of VL training.

4 Experiments and Results

We hypothesize that grounded pretraining leads models to learn better linguistic representations than does text-only pretraining. Specifically, we are interested in whether grounded pretraining yields benefits on NLP tasks that are defined entirely over textual inputs, so do not *require* grounded representations (i.e., as opposed to tasks like visual question answering, for which the need for grounded representations is not debatable). We consider three different evaluations of "semantics": commonsense reasoning about the physical world (§4.1), inferring sentence-level semantic structure (§4.2), and composing lexical semantic concepts (§4.3).

4.1 Physical Commonsense QA

We first ask whether VL pretraining yields gains to benchmark NLP tasks that intuitively rely on multimodal knowledge, even if they don't explicitly require representing non-text inputs. We use PhysicalQA (PIQA) (Bisk et al., 2020b), a commonsense reasoning benchmark in which models are given a sentence describing a physical goal (*"Remove gloss from furniture."*) and must select between two candidate solutions (*"Rub furniture with steel*

wool."/ "Rub furniture with cotton ball."). Following the setup in PIQA, we consider each solution candidate independently by combining the goal with one solution (`[CLS]` goal `[SEP]` solution `[SEP]`), and using the `[CLS]` token embedding at the last hidden layer as the representation of the candidate. We train a probing classifier to perform a binary classification task, with the two candidate representations as its inputs.

We consider linear, MLP, and transformer probing classifiers. For the linear and MLP probes, we freeze the encoder weights and only train the classifiers. For the transformer probe, we finetune the last transformer encoder layer and a linear layer on top of it. See Appendix B.1 for details.

Table 1 shows our results. Across all settings, we see that VL pretraining produces consistent but marginal gains. In addition, we see that training on YouTube video captions, even without using the video information itself (e.g., comparing VideoBERT$_{text}$ to original BERT) yields a few-point improvement. Figure 1 shows the results based on word-level edit distance between two solutions. We see that VL pretraining brings a few points improvements when edit distance is low (one or two words), i.e., where picking the right solution hinges on grounded information for single lexical items. On manual inspection of the errors, we do not observe any consistent patterns that reflect different behaviors for VL models and text models. This is true even when we focus only on cooking-related examples for VideoBERT models (i.e., examples we expect to be in domain and thus most likely to demonstrate gains).

Thus, overall, our results are mixed. We see that VL pretraining can yield improvements on text-only tasks, and that these gains likely come from both the difference in the distribution of language as well as the non-linguistic information itself. However, the gains are quite small–only a few points, despite the fact that the task in question (PIQA) is intended to directly probe the type of understanding that one gains from interacting with the physical world. We note, however, that most of the goals and solutions in PIQA are not cooking-related, and thus the limited impact might be due to domain mismatch. Future work on domain-general VL pretraining would offer valuable insight.

4.2 Coreference and Semantic Roles

The PIQA results above suggest VL pretraining yields some gains on extrinsic tasks like QA. Such

Encoder	Linear	MLP	Trans.
BERT_base	55.43 ± 0.31	57.98 ± 0.16	60.12 ± 1.43
VideoBERT_text	57.87 ± 0.64	58.97 ± 0.44	62.35 ± 1.23
VideoBERT_VL	58.51 ± 0.20	58.56 ± 0.27	63.66 ± 1.31
VisualBERT_text	54.81 ± 0.19	56.81 ± 0.24	58.63 ± 0.79
VisualBERT_VL	55.83 ± 0.27	59.10 ± 0.11	61.66 ± 1.08

Table 1: Accuracy $\pm$ std. of different pretrained representations on the validation split of PIQA. Numbers are averaged over five runs. VL pretraining only brings marginal improvements over text-only pretraining.
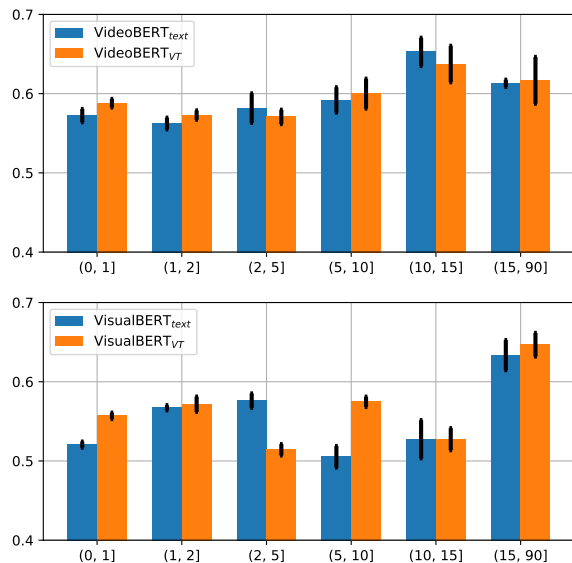

Figure 1: Accuracy divided by word-level edit distance between two solutions. Results are averaged over five runs. Standard deviations are represented by the black lines. VL pretraining brings limited improvements over text-only pretraining on samples with low edit distance.

findings invite questions of whether the gains are driven by intrinsic improvements in the semantic representations themselves. Semantics, after all, is about building intermediate representations that enable the surface form of natural language to ground to entities and events in the real world. Thus, intuitively, one might expect that training with explicit access to entities and events would improve such representations (e.g., of predicate-argument structure or entity coreference). To test this intuition, we use the edge probing framework from Tenney et al. (2019b), in which a probing classifier takes as input a token span(s), represented as a weighted sum of the layer activations of the token embeddings in the words, and needs to predict a task-related label (e.g. part of speech, parse information). The evaluation suite includes ten syntactic and semantic tasks. Results for all tasks, along with training

details, are given in Appendix B.2. Per the above intuition, we are particularly interested in tasks that probe semantic structure. We focus on the following: Entity Coreference (Coref.), e.g., recognizing that *"apples"* and *"them"* refer to the same entity in *"After the apples are chopped, put them in the bowl"*; Semantic Role Labeling (SRL), which requires encoding semantic agents and patients, e.g., recognizing that *"carrots"* are the recipient of the pureeing action in *"The carrots are then pureed in the food processor"*; Semantic Proto-Roles (SPR), which requires predicting features such as *awareness* or *cause* for words in context, e.g., recognizing *"the food processor"* causes the pureeing event, but is not aware of it; and Semantic Relations (Rel.), which requires predicting relations like entity-destination, e.g., the relation between *"apples"* and *"bowl"* in *"put the apples in the bowl"*.

Table 2 shows results. Across the board, we observe extremely marginal gains in performance when comparing VL models to their text-only counterparts. In 7 out of 8 comparisons, the VL model outperforms the text model, versus just 1 comparison in which the text model outperforms. However, the differences that exist do not appear meaningful (~ 0.5 percentage points), and we thus do not conclude that VL pretraining leads to any clear improvement in the models' ability to encode abstract semantic structure.

Encoder	SRL	Coref.	SPR	Rel.
BERT_base	90.10 ± 0.20	95.90 ± 0.00	83.70 ± 0.00	76.25 ± 0.05
VideoBERT_text	84.33 ± 0.05	92.47 ± 0.05	78.23 ± 0.05	65.83 ± 0.21
VideoBERT_VL	84.73 ± 0.05	92.82 ± 0.05	78.80 ± 0.00	66.37 ± 0.80
VisualBERT_text	89.00 ± 0.00	94.87 ± 0.05	82.27 ± 0.05	74.37 ± 0.19
VisualBERT_VL	89.57 ± 0.21	95.13 ± 0.05	82.17 ± 0.09	74.83 ± 0.05

Table 2: Comparison of models encoding of various aspects of sentence-level semantic structure (average of micro-averaged F1 score of three runs). We see no significant improvements from VL pretraining. Decreases in performance relative to BERT_base are likely due to the domain mismatch between image/video captions and the probing evaluation sets (newswire/web text).

4.3 Adjective-Noun Composition

Finally, we investigate whether multimodal pretraining impacts conceptual structure at the lexical level. Arguably, if VL pretraining were to affect linguisitic representations in any meaningful way, we would expect it to manifest in the conceptual representations of visually-groundable concepts. To explore this, we focus on adjective-noun compo-
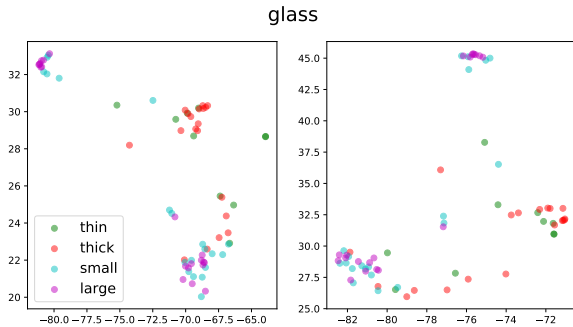
Figure 2: TSNE projections of noun representations clusters for "glass" of VideoBERT_text (Left) and VideoBERT_VL (Right). Neither model distinguishes *small / large* or *thin / thick*.

Encoder	Homo.	Compl.	V-Meas.
BERT_base	0.500 ± 0.017	0.526 ± 0.020	0.513 ± 0.018
VideoBERT_text	0.684 ± 0.025	0.702 ± 0.025	0.693 ± 0.025
VideoBERT_VL	0.663 ± 0.016	0.678 ± 0.016	0.670 ± 0.016
VisualBERT_text	0.528 ± 0.025	0.546 ± 0.028	0.537 ± 0.027
VisualBERT_VL	0.546 ± 0.021	0.571 ± 0.022	0.558 ± 0.022

Table 3: Summary metrics (range 0 to 1) for clustering noun embeddings (e.g., "apple") according to their adjective modifiers (e.g., "ripe"). Numbers are averaged over five random seeds. We see no significant improvement in any metric when grounded (video or image) data is included during training. Homogeneity of 1 means that every point in a cluster belongs to the same class. Completeness of 1 means that every point belonging to a given class is in the same cluster. V-measure is the harmonic mean of the two.

Encoder	Accuracy
BERT_base	0.968 ± 0.002
VideoBERT_text	0.992 ± 0.001
VideoBERT_VL	0.993 ± 0.001
VisualBERT_text	0.984 ± 0.002
VisualBERT_VL	0.982 ± 0.001

Table 4: Results of probing the noun embeddings to classify the adjectives that modify the nouns. Numbers are averaged over five runs.

sition, as this provides a simple way of defining a space of visually-groundable objects and properties that we expect conceptual representations to encode. For example, we expect that embeddings of the word "knife" from contexts in which the knife is described as "sharp" should be more similar to other instances of sharp knives than to instances of knives that are described as "dull".

We focus on the list of visually grounded adjec-

tives introduced in Isola et al. (2015) (e.g., "small", "bright", "sharp"). We then mine the WikiHow dataset (Koupaee and Wang, 2018) for all adjective-noun bigrams involving these adjectives. We chose WikiHow because it does not overlap with the training corpus of either of our models, but contains similarly concrete, descriptive language. We perform several additional filters to remove low frequency bigrams, described in Appendix B.3, which results in an analysis set of 651 unique adjective-noun bigrams across 11,970 contexts. We test how well each pretrained model's representations of the noun (e.g., "knife") encodes information about the adjective (e.g., "sharp") that modified it.

Figure 2 provides a qualitative example of how noun representations cluster when using representations from VideoBERT_text vs. VideoBERT_VL. Quantitatively (Tables 3 and 4; see Appendix B.3 for experimental details), we do not see significant differences between VL and text-only models. Thus, again, VL pretraining does not appear to produce the desired improvements.

5 Conclusion

We provide a series of experiments which compare grounded vision-and-language (VL) pretraining to comparable text-only pretraining in terms of the quality of the linguistic representations produced. We find that VL pretraining sometimes produces gains, but that the text-only baselines perform well, and thus the margins are too small to support conclusions that VL pretraining (in its current form) has benefits for NLP in general. While there are good arguments to be made that grounding is necessary for learning general-purpose language representations, we conclude that current methods, which use direct extensions of NLP architectures and are often trained on data from narrow domains, have yet to produce such benefits. Future work is required to explore more domain-general VL training, as well as alternative architectures and losses for combining vision and language signals.

Acknowledgments

References

Emily M. Bender and Alexander Koller. 2020. Climbing towards NLU: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5185–5198, Online. Association for Computational Linguistics.

Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, Nicolas Pinto, and Joseph Turian. 2020a. Experience grounds language. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8718–8735, Online. Association for Computational Linguistics.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020b. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7432–7439.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *CoRR*, abs/2005.14165.

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

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. 2015. Microsoft COCO captions: Data collection and evaluation server. *CoRR*, abs/1504.00325.

Gabriella Chronis and Katrin Erk. 2020. When is a bishop not like a rook? when it's like a rabbi! multi-prototype BERT embeddings for estimating semantic relationships. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 227–244, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Maxwell Forbes, Ari Holtzman, and Yejin Choi. 2019. Do neural language representations learn physical commonsense? *CoRR*, abs/1908.02899.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.

Steve R Howell, Damian Jankowicz, and Suzanna Becker. 2005. A model of grounded language acquisition: Sensorimotor features improve lexical and grammatical learning. *Journal of Memory and Language*, 53(2):258–276.

Phillip Isola, Joseph J Lim, and Edward H Adelson. 2015. Discovering states and transformations in image collections. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1383–1391.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2019. SpanBERT: Improving pre-training by representing and predicting spans. *arXiv preprint arXiv:1907.10529*.

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Mahnaz Koupaee and William Yang Wang. 2018. Wikihow: A large scale text summarization dataset. *CoRR*, abs/1810.09305.

Angeliki Lazaridou, Nghia The Pham, and Marco Baroni. 2015. Combining language and vision with a multimodal skip-gram model. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 153–163, Denver, Colorado. Association for Computational Linguistics.

Yoav Levine, Barak Lenz, Opher Lieber, Omri Abend, Kevin Leyton-Brown, Moshe Tennenholtz, and Yoav Shoham. 2020. Pmi-masking: Principled masking of correlated spans.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2020. What does BERT with vision look at? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5265–5275, Online. Association for Computational Linguistics.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Arjun Majumdar, Ayush Shrivastava, Stefan Lee, Peter Anderson, Devi Parikh, and Dhruv Batra. 2020. Improving vision-and-language navigation with image-text pairs from the web. In *European Conference on Computer Vision*, pages 259–274. Springer.

William Merrill, Yoav Goldberg, Roy Schwartz, and Noah A. Smith. 2021. Provable limitations of acquiring meaning from ungrounded form: What will future language models understand? *CoRR*, abs/2104.10809.

Alec Radford, Jong Wook Kim, Chris Hallacy, A. 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 *ICML*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Andrew Rosenberg and Julia Hirschberg. 2007. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pages 410–420.

Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019. VideoBERT: A joint model for video and language representation learning.

Hao Tan and Mohit Bansal. 2020. Vokenization: Improving language understanding with contextualized, visual-grounded supervision. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2066–2080, Online. Association for Computational Linguistics.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019a. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

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. 2019b. What do you learn from context? probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations*.

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

Ivan Vulić, Edoardo Maria Ponti, Robert Litschko, Goran Glavaš, and Anna Korhonen. 2020. Probing pretrained language models for lexical semantics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7222–7240, Online. Association for Computational Linguistics.

Lang Yu and Allyson Ettinger. 2020. Assessing phrasal representation and composition in transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4896–4907, Online. Association for Computational Linguistics.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

A VL and Text-only Pretraining

A.1 Domain-specific Masking

Masking tokens uniformly at random in BERT is found to be suboptimal (Joshi et al., 2019; Levine et al., 2020). In addition, we hypothesize that the benefits of visual-linguistic alignment might be greater if masking occurs on content words (which, in the cooking domain, are likely to be visually-groundable concepts). Thus, we implement a *domain-specific masking*, which aggressively masks the most frequent cooking-related verbs and nouns. We apply the BERT tokenizer on the cooking corpus, and manually pick the most frequent 500 cooking-related tokens. During the pretraining data generation, 15% of the tokens will be chosen, where the frequent tokens has 80% probability of being chosen, while the other tokens has 15% probability. The masking strategy is similar to the original BERT, where 80%/10%/10% of the chosen tokens will be replaced with `[MASK]` tokens/ random tokens/ the original tokens respectively. VideoBERT is pretrained with both random masking and domain-specific masking, while VisualBERT is only pretrained with random masking.

A.2 Pretraining Details

We pretrain VideoBERT$_{\text{text}}$ from scratch on the same cooking dataset in (Sun et al., 2019). We strictly follow the training setup of VideoBERT$_{\text{VL}}$ which is based on BERT$_{\text{base}}$: it has 12 layers of transformer blocks, where each block has 768 hidden units and 12 self-attention heads. We use 4 Cloud TPUs with a total batch size of 128, and we train a model for 400K iterations. We use the Adam optimizer with an initial learning rate of 1e-5, and a linear decay learning rate schedule. The training process takes around 2 days.

We initialize VisualBERT$_{\text{text}}$ with the pretrained BERT$_{\text{base}}$ weights released by (Devlin et al., 2018). This text-only model has the same configuration as its VL variant: it has 12 layers of transformer blocks, where each block has 768 hidden units and 12 self-attention heads. The training process also largely follows the setup of VisualBERT$_{\text{VL}}$: we use 4 TitanV GPUs with a total batch size of 64 and cap the sequences whose lengths are longer than 128. VisualBERT$_{\text{text}}$ is trained for 10 epochs, or roughly 90K iterations, with the Adam optimizer with an initial learning rate of 5e-5. The warm-up step number is set to 10% of the total training step count. The training process takes around 25 hours.

B Experimental Details

B.1 PIQA

We use `[CLS]` token embedding e at the last hidden layer as the representation of a candidate solution for a goal. This embedding will be passed into the probing classifiers: a single linear layer, an MLP, and a transformer. The MLP probe has a hidden size of 512 and has architecture as below:

$$h = \texttt{tanh}(W_1 e + b_1)$$
$$h = \texttt{layer\_norm}(h)$$
$$\text{output} = W_2 h + b_2$$

We train a model by cross-entropy loss and by using the Adam optimizer (Kingma and Ba, 2014) with a batch size of 32, an initial learning rate of 1e-4. We evaluate a model on the validation set every 1000 steps, halve the learning rate if no improvement is seen in 5 validations, and stop training if no improvement is seen in 20 validations. In this way, we limit the expressive power of the probes (since we are primarily interested in understanding differences in the representations that result directly from pretraining), yet still consider a number of ways (linear/nonlinear) that such information could potentially be encoded.
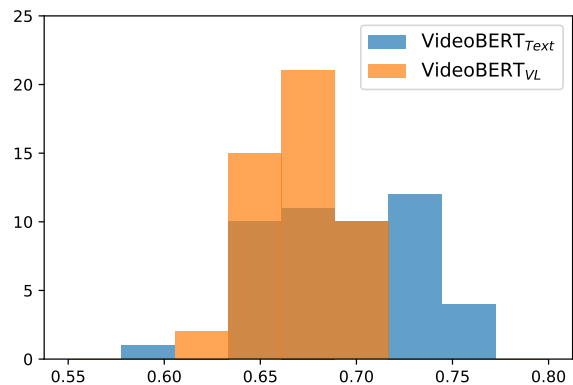
Figure 3: Histograms of V-measure scores for clustering noun embeddings according to their adjective modifiers of VideoBERT$_{\text{text}}$ (Random) and VideoBERT$_{\text{VL}}$ (Random).

B.2 Syntactic and Semantic "Edge Probing" Tasks

Edge probing formulates probing tasks into the same format, where the probing classifier takes a span $s_1 = [i_1, j_1)$ and an optional span $s_2 = [i_2, j_2)$, and needs to predict a task-related label

Encoder	Masking	Batch Size	32			64			128		
		LR	1e-4	5e-5	2e-5	1e-4	5e-5	2e-5	1e-4	5e-5	2e-5
BERT$_{base}$	Random		60.12 ± 1.43	60.61 ± 1.82	58.08 ± 1.34	59.17 ± 1.37	58.14 ± 1.31	57.21 ± 0.39	59.58 ± 1.57	58.18 ± 0.75	56.79 ± 0.30
VideoBERT$_{text}$	Random		62.02 ± 0.73	60.10 ± 1.07	58.45 ± 1.63	61.35 ± 1.31	60.89 ± 0.58	57.94 ± 0.91	61.06 ± 0.95	59.89 ± 0.50	56.40 ± 0.51
VideoBERT$_{VL}$	Random		63.66 ± 1.31	62.68 ± 0.76	60.51 ± 0.66	62.77 ± 0.80	63.13 ± 0.35	59.60 ± 0.96	62.59 ± 0.66	61.39 ± 0.40	59.65 ± 0.73
VideoBERT$_{text}$	Domain		61.66 ± 0.64	60.04 ± 0.57	58.97 ± 0.63	60.66 ± 0.87	60.42 ± 1.13	58.43 ± 1.15	60.18 ± 0.98	58.48 ± 0.53	57.62 ± 1.58
VideoBERT$_{VL}$	Domain		62.30 ± 0.95	61.10 ± 0.43	59.67 ± 0.40	61.73 ± 0.53	60.67 ± 0.74	59.26 ± 0.46	61.48 ± 0.60	59.99 ± 1.21	58.03 ± 1.18
VisualBERT$_{text}$	Random		58.63 ± 0.79	57.95 ± 0.71	56.09 ± 1.20	58.65 ± 0.86	57.69 ± 0.56	55.32 ± 0.75	58.03 ± 1.16	57.81 ± 0.83	55.51 ± 0.90
VisualBERT$_{VL}$	Random		61.66 ± 1.08	60.78 ± 0.38	59.30 ± 0.96	60.70 ± 0.55	60.55 ± 0.66	59.09 ± 0.79	60.40 ± 0.72	60.07 ± 0.18	58.52 ± 1.04

Table 5: Hyperparameter search for transformer probing models on PIQA. LR refers to learning rate. Numbers are average accuracy of five runs. VL checkpoints consistently outperform text-only counterparts by a small margin. This rules out the likelihood that the improvement of VL pretraining is caused by specific hyperparameter settings.

Encoder	Masking	POS	Consti.	Dep.	NER	SRL	Coref.^{O}	SPR1	SPR2	Coref.^{W}	SemEval
BERT$_{base}$	Random	96.47 ± 0.05	86.80 ± 0.14	95.20 ± 0.08	96.00 ± 0.00	90.10 ± 0.20	95.90 ± 0.00	83.70 ± 0.00	82.80 ± 0.08	57.90 ± 0.00	76.25 ± 0.05
VideoBERT$_{text}$	Random	93.90 ± 0.28	82.30 ± 0.14	92.07 ± 0.05	91.60 ± 0.14	84.33 ± 0.05	92.47 ± 0.05	78.23 ± 0.05	81.30 ± 0.00	56.20 ± 0.14	65.83 ± 0.21
VideoBERT$_{VL}$	Random	93.87 ± 0.09	83.50 ± 0.00	92.27 ± 0.05	92.00 ± 0.14	84.73 ± 0.05	92.82 ± 0.05	78.80 ± 0.00	81.30 ± 0.00	56.20 ± 1.98	66.37 ± 0.80
VideoBERT$_{text}$	Domain	93.10 ± 0.00	82.20 ± 0.00	91.10 ± 0.16	90.77 ± 0.33	82.93 ± 0.05	92.40 ± 0.00	76.83 ± 0.12	81.03 ± 0.05	54.43 ± 0.90	62.97 ± 0.47
VideoBERT$_{VL}$	Domain	93.33 ± 0.09	82.37 ± 0.09	91.47 ± 0.09	90.67 ± 0.17	82.93 ± 0.05	92.30 ± 0.00	78.03 ± 0.05	81.13 ± 0.09	56.07 ± 0.52	64.63 ± 0.19
VisualBERT$_{text}$	Random	95.40 ± 0.00	86.20 ± 0.14	94.20 ± 0.00	94.60 ± 0.57	89.00 ± 0.00	94.87 ± 0.05	82.27 ± 0.05	82.40 ± 0.00	57.57 ± 2.03	74.37 ± 0.19
VisualBERT$_{VL}$	Random	96.10 ± 0.14	86.23 ± 0.05	94.57 ± 0.05	95.20 ± 0.00	89.57 ± 0.21	95.13 ± 0.05	82.17 ± 0.09	82.43 ± 0.05	58.13 ± 0.61	74.83 ± 0.05

Table 6: Results of models on a suite of edge probing tasks. Numbers reported are average of micro-averaged F1 score of three runs. We see no significant improvements when the models are pretrained with both text and visual data.

Encoder	Masking	Homo.	Compl.	V-Meas.
BERT$_{base}$	Random	0.500 ± 0.017	0.526 ± 0.020	0.513 ± 0.018
VideoBERT$_{text}$	Random	0.684 ± 0.025	0.702 ± 0.025	0.693 ± 0.025
VideoBERT$_{VL}$	Random	0.663 ± 0.016	0.678 ± 0.016	0.670 ± 0.016
VideoBERT$_{text}$	Domain	0.409 ± 0.014	0.442 ± 0.013	0.425 ± 0.014
VideoBERT$_{VL}$	Domain	0.393 ± 0.031	0.422 ± 0.031	0.407 ± 0.031
VisualBERT$_{text}$	Random	0.528 ± 0.025	0.546 ± 0.028	0.537 ± 0.027
VisualBERT$_{VL}$	Random	0.546 ± 0.021	0.571 ± 0.022	0.558 ± 0.022

Table 7: Summary metrics (range 0 to 1) for clustering noun embeddings (e.g., "apple") according to their adjective modifiers (e.g., "ripe"). Numbers are averaged over five random seeds. We see no significant improvement in any metric when grounded (video or image) data is included during training. Homogeneity of 1 means that every point in a cluster belongs to the same class. Completeness of 1 means that every point belonging to a given class is in the same cluster. V-measure is the harmonic mean of the two.

Encoder	Masking	Accuracy
BERT$_{base}$	Random	0.968 ± 0.002
VideoBERT$_{text}$	Random	0.992 ± 0.001
VideoBERT$_{VL}$	Random	0.993 ± 0.001
VideoBERT$_{text}$	Domain	0.964 ± 0.002
VideoBERT$_{VL}$	Domain	0.973 ± 0.001
VisualBERT$_{text}$	Random	0.984 ± 0.002
VisualBERT$_{VL}$	Random	0.982 ± 0.001

Table 8: Results of probing the noun embeddings to classify the adjectives that modify the nouns. Numbers are averaged over five runs.

based on the span representations. A span representation is a weighted sum of the layer activations of the token embeddings in the given spans. We train a probing classifier for each task with encoder weights frozen, and follow the probing architecture and training strategy in (Tenney et al., 2019c). Figure 6 shows the results on all tasks for all models.

B.3 Lexical Composition

We preprocess WikiHow dataset by tokenizing the 215K instructions into 5 million single sentences. We run a bigram search over all the sentences to find pairs of an adjective and a noun. The lower bound of bigram occurrence is set to 10, while the bigrams whose nouns do not pair with more than 10 unique adjectives are filtered out. Eventually, this leaves us 57,521 bigrams and 651 unique bigrams. Encoders then produce the representations of the nouns in these bigrams.

Following, we apply a visually grounded adjective filter based on the list of adjectives introduced in (Isola et al., 2015). For a unique bigram, up to 20 noun representations are randomly sampled. Finally, there are 62 unique adjectives, 48 unique nouns, and 11,970 noun representations.

We use K-Means to cluster the representations of each noun, with K equal to the the number of unique adjectives that modifies the noun in our dataset. We measure the quality of the resulting clusters using three clustering metrics: homogene-

ity, completeness, and V-measure [2](Rosenberg and Hirschberg, 2007), which are roughly analogous to precision, recall, and f1-score. We use the adjectives as the ground-truth cluster labels; i.e., scores are higher when the noun representations cluster according to the adjectives which modifies the noun in context.

Last, we carry out a probing experiment to attempt to evaluate the adjective information that is linearly encoded in the noun representations produced by the models. Given noun embeddings, a linear probing classifier, that is built on top of each model, classifies the adjectives that modify the nouns.

Based on a series of quantitative analyses, Tables 7 and 8 and Figure 3, we do not see significant differences between VL and text-only models.

---

[2]The ratio of weight attributed to homogeneity and completeness is set to 1:1.