JointXplore: Testing and Exploring Joint Visual-NLP Networks

Leonard Schenk
Department of Computer Engineering
Bogazici University Istanbul / Technical University of Munich
leonard.schenk@tum.de

Abstract

In the field of software engineering, testing is a crucial aspect, especially when dealing with probabilistic, potentially safety-critical neural networks. Despite the importance of testing multimodal networks on tasks such as Visual Question Answering (VQA), limited research has been conducted in this area [15, 30]. In an effort to address this gap in the literature, this study presents three experiments that assess the accuracy, coverage, and robustness of two multimodal neural network architectures on VQA tasks. Additionally, we examine the performance of these architectures when using only the textual input for VQA. The results of the experiments indicate that both architectures exhibit relatively high performance when using text alone. However, coverage metrics reveal that the text input alone leads to the discovery of fewer internal states compared to the combined vision-language input. Finally, the use of state-of-the-art adversarial attack methods highlights the vulnerability of multimodal neural networks. The codebase for these experiments and the results are freely available at Github.

1. Introduction

In recent years, the use of multi-modality tasks in Deep Learning has gained significant attention [5, 7]. These tasks involve the creation of deep learning models that can process multiple types of input, such as visual and linguistic information, and use the combined features to produce meaningful output. One area of focus within this field is the integration of visual and linguistic information in tasks such as Image Captioning [5, 7], Visual Grounding [1, 4, 5, 26], Visual Entailment [6, 20, 31], and Visual Question Answering [3, 9, 28]. This research paper will specifically focus on the latter task, Visual Question Answering (VQA), as it is considered a crucial element in human-machine interaction.

While Deep Neural Networks (DNNs) have achieved impressive results on various tasks, they are not immune to failure, particularly in safety critical applications where even a single error is not tolerable [14, 24]. In addition, the vulnerability of DNNs to adversarial attacks, which are designed to trick the network into making incorrect predictions [11, 17], has further highlighted the importance of thoroughly testing and verifying the robustness and resilience of these models. In response to these concerns, a field of research has emerged focused on measuring the robustness and resilience of DNNs, with a particular focus on computer vision models [11, 22, 24, 32]. However, there has been relatively less work on fault detection in natural language processing [30] and almost none on multimodal neural networks, such as VQA systems [15]. This lack of research in these areas may be due to the popularity of visual tasks, as well as the challenge of perturbing non-visual modalities without losing interpretability. In this paper, we aim to address this imbalance by exploring methods of testing VQA systems. Our contributions include:

- Determining vision-language input dependencies by experimenting with turning off the visual modality.
- Analyzing multiple coverage metrics [22] for two different architectural approaches for VQA models.
- Assessing the vulnerability of VQA system with respect to adversarial examples.

2. Related Work

Testing for DNNs. Concepts of testing conventional software are hard to transfer to DNNs as of the fundamental differences in the program architecture. Generally, traditional software code has many different branches, which are only executed if a certain condition is fulfilled. In neural networks however, every line of code tends to participate in the program execution, regardless of the input. Thus, traditional metrics like coverage lose their meaning with DNNs. However, Pei et al. [24] introduced a way of measuring coverage for DNNs by looking at how many neurons are activated over a certain threshold for a given input. This metric was later refined [22] and used to facilitate automated, coverage-guided testing and input generation [32].

However, besides some work about robust multimodal networks [15], there has not been made much effort into testing multimodal systems, especially for vision-language downstream tasks like VQA.

Visual Question Answering. In general, the task of VQA is to provide answers to questions that are thematically related to the content of an image. First research in this field was published in 2015 [3] and 2016 [9]. Afterwards, research in this field gained more popularity, especially with the release of the benchmark VQA 2.0 [12]. In the beginning, most model architectures consisted of different modules for extracting image and language features before merging both modalities [33]. However, after the rise of the transformers [27] models emerged that combine the tokenized vision and language input directly in a multimodal transformer [16]. In general, the task of VQA can be viewed as a classification problem where the model picks the best answer among a set of candidates [16]. Alternatively, other approaches use seq2seq models to create new sentences as answer output [19]. The current state of the art in this field is the general-purpose multimodal model BEiT-3 [28], which performs pre-training on a large amount of data and transfers one multiway transformer model to handle various tasks. For our studies, we will compare the pure vision-language transformer architecture *ViLT* [16] with the method *ALBEF* [18]. The latter first aligns image and text input as a pre-training objective before it fuses both modalities with a transformer-based architecture.

3. Data

For the following experiment, the VQA 2.0 dataset [12] is used, which is based on the images of the *COCO* [21] dataset. The training and validation splits contain 443 757/214 354 questions and annotations on 82 783/40 503 images, respectively. However, due to the lack of computational resources and since only evaluation and no training is performed, a smaller subset is deemed sufficient. Specifically, 5 000 question/annotation pairs from the training set are used for determining the initial coverage regions, while 2 500 question/annotation pairs are used for computing the actual coverage values on each model. It should also be noted, that each question has ten answers in total, where some of them may be the same or different from each other. Thus, it makes sense to weigh the answers according to their frequency for the loss computation. The authors of the dataset proposed a standardized way for doing this by computing the accuracy of an answer as

$$Acc(ans) = min\{\frac{\#\text{humans that said ans}}{3}, 1\}.$$

4. Method

4.1. Task

In this study, we examine the multimodal task of Visual Question Answering (VQA), in which an image and corresponding textual question are input and a predicted answer is generated by fusing both modalities. Previous research has characterized the VQA process as both a classification and seq2seq task, and thus this work aims to explore both possibilities. Additionally, this study focuses on investigating the interaction between the visual and linguistic modalities in the VQA process. The following research questions are proposed:

4.2. Research Questions

DNNs can only correctly master the task of VQA if they take into account the information of both the visual and the textual domain [29]. In fact, there exists a risk that especially classification-models are prone to overfitting on the limited set of answers and predicting the answer solely from the question [2]. This leads to the first research question

- **RQ1 (Single-Modality Performance):** What is the change in performance for VQA models when they are forced to predict the answer only based on the question? Can they still guess the correct answer without any meaningful image information? Do seq2seq models have a higher decrease in performance, as they are not trained to predict a limited set of answers?

While the performance of the model gives a first overview about the impact of switching off one modality, metrics such as coverage promise to give more insights into the behaviour of a DNN under certain conditions [22, 24]. With this in mind, we arrive at

- **RQ2 (Coverage):** How does switching off one modality affect the coverage of the models? Does the network only activate a subset of the previous activated neurons? Or are new regions coverage regions explored?

Finally, another important aspect of testing neural networks is the robustness against adversarial examples (AEs) [11, 17]. While the research about AEs mainly focused on the visual domain, recent results have shown that AEs can also be generated for language-based tasks [10, 34]. Using this, we can formulate

- **RQ3 (Adversarial text):** How robust are VQA models with respect to perturbation of the textual part of the input? Does the robustness degrade if the question is the only input, i.e. the visual input is turned off? Is either one of the classification or seq2seq2 architecture superior to the other one in terms of robustness?

4.3. Models

Firstly, to obtain answers to the previously defined research question, two models are needed that fit the description of a classification and a seq2seq model. As classification model, *ViLT* [16] is used. In detail, *ViLT* first tokenizes image and question seperately and feeds the tokenized version of both entities into a multimodal transformer encoder. Afterwards, the multimodal features are aggregated in a pooler module, whose output is used as the input to a final, fully-connected, multi-layer perceptron for classification [16]. A visual representation of the architecture can bee seen in 1a.

In contrast, *ALBEF* [19] conducts a more thorough preprocessing step to align image and text input. To achieve this, a contrastive pre-training objective is formulated that promotes a high similarity score between the encodings of corresponding question-image pairs and a low similarity score between negative pairs. After that, both modalities are fused in a multi-modal transformer encoder. During training, this process is supported by a teacher model, which achieves ensemble-like knowledge distillation in the network [13]. To obtain the final answer, a transformer decoder is added on top of the encoder, which generates the answer word by word. This makes the possible answer space much larger and thus, it is way harder to predict exactly the right answer. To accommodate this inequality, the decoder will output the answer-candidate from the classification problem that has the highest similarity to the generated answer [19]. This technique establishes a fair ground for comparing both methods. The high-level architecture of *ALBEF* can be seen in fig. 1b.

4.4. Experimental Design and Setup

In order to answer *RQ1*, a benchmark is established for both models. A subset of the evaluation split, as described in 3, is used to compute the accuracy for both models with full images enabled. Consecutively, in the next run the same questions are evaluated with random images. This is done by setting each pixel in each channel of the original rgb image to a random value between 0 and 255. The resulting difference in accuracy between running the model with and without images, gives a measure for how much the model predicts the answer purely based on the question and how much it actually uses the image information. However, it is important to note that weights for *ViLT* are only available after it was trained on both, the train and the val split. This unfortunately decreases comparability between the two models, as *ViLT* has already seen the val split. Nevertheless, by comparing the accuracy of the same *ViLT* model once with full and once with random images, the intuition behind the discrepancy of the two different accuracy scores stays the same.

Considering *RQ2* and following the original approach

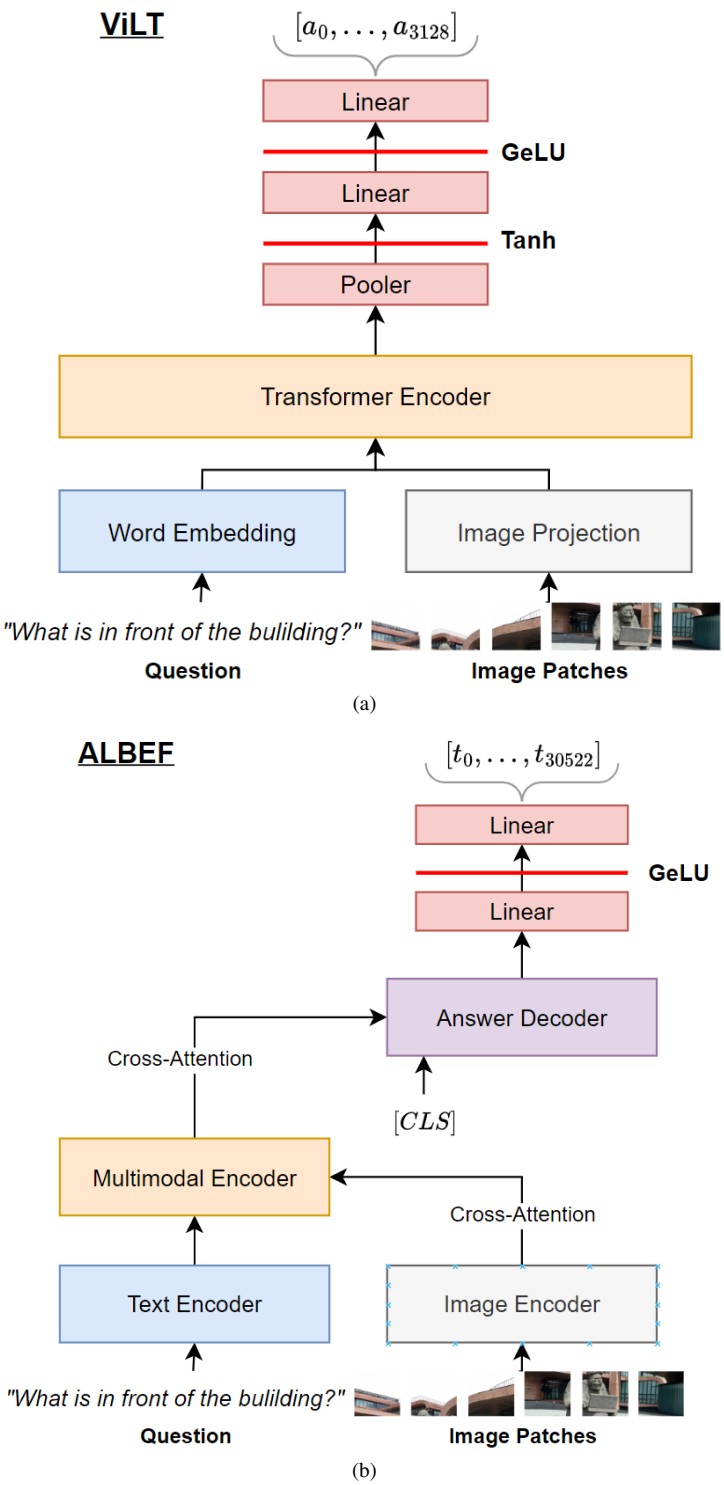

Figure 1. (a) The vision language transformer *ViLT* [16] computes an inexpensive embedding and projection for both modalities, respectively. A transformer encoder extracts the multimodal features. After pooling and after a linear layer there are two activation functions, from where the neuron coverage is measured. (b) As for (a) but with the *ALBEF* [19] model. Question and Image are encoded and fused in a multimodal encoder. These featues are fed into an Answer Decoder with cross-attention. Finally, two linear layers determine the model output with a GeLU activation in between for coverage measurement.

of *DeepXplore* [24], for each selected neuron n_i, boundary values have to be calculated. This is done by saving n^i_{min} and n^i_{max} while performing VQA on a subset of the train split as specified in 3. Here, n^i_{min} is the smallest value that the i-th neuron assumes for all inputs in the set. n^i_{max} is the same for the maximum value, respectively. These boundaries represent the main activation regions for each neuron during training. Using these values as a reference, the validation split is used to compute *K-Section*-, *Boundary*- and *Strong-Neuron*-coverage. Intuitively, *K-Section*-coverage measures the extent of the neurons' activation in the main region, whereas the corner case behaviour of the network is displayed by the latter two metrics. Thus, these fine-grained metrics are more suited to develop a deeper insight into the networks' behaviour, compared to standard neuron coverage. For a more detailed, mathematical definition, the reader is referred to the original paper *DeepGauge* [22]. So far, the approach from above allows us to calculate coverage for any neuron n^i in the network. Computing the coverage for every neuron in the network would require many ressources and would fail to address the regions of interest. Specifically, we are interested in the neuron values after an activation function, as these points introduce the non-linearities and it is decided if a neuron is active or not. Moreover, we only use layers closer to the output to emphasize these layers, as they have generally more impact on the final outcome. In fig. 1b and fig. 1b, red lines labeled with an appropriate activation function show how the coverage layers were chosen for each of the two models. Finnaly, to answer *RQ3*, we leverage the *textattack* [23] library to create adversarial text examples for both models. *textattack* combines a collection of different attack strategies and allows its users to test these strategies on any NLP-network. Since VQA is of the Visual-NLP-domain, a wrapper is created for each model that takes the perturbed text and feeds it together with the image to the network. For *ViLT*, *Probability Weighted Word Saliency* [25] was used as an untargeted classification attack method. It swaps synonyms based on predetermined indicators such as saliency score and swap-effectiveness. Since *ALBEF* is a seq2seq model, a different attack strategy had to be applied. *Seq2sick* [8] fits this requirement well as it crafts adversarial examples for seq2seq models by solving an optimization problem. With the attack methods selected, 80 questions are evaluated on both models with full and with random images. The evaluation is based on the success rate of the attack and the effort it took to fool the network. For fairness and due to high computational costs, both methods were limited to produce a maximum of 30 different input texts before the attack was classified as failed. A notion of effort can be obtained by measuring the success rate and the average number of attempts until success or failure is returned.

5. Results

5.1. RQ1: Single-Modality Performance

Table 1 shows the results of evaluating both models with full and with random images. It becomes clear that discarding the image information drastically decreases the performance, as both models' accuracy drops by 44.05% and 54.12%, respectively. As for *ViLT*, the table shows that the model can still guess the correct answer for roughly every second question. Since no question in the dataset is actually answerable without the image, this is a remarkably high amount compared to the statistical chance of guessing correct, which is $\frac{1}{3129} = 0.032\%$. However, it is hard to interpret this result correctly. Either, it could show that the model can effectively narrow down the most probable answer candidates and exclude nonsensical possibilities to increase its chance of making a good guess from the remaining candidates. On the other hand, it could mean that the model was overfit on the questions and does not even need the image to answer almost half of them correctly. A similar behaviour is found for *ALBEF*. The main difference is that ALBEF showcases a lower performance on random images and a larger performance drop on random ones. Thus, compared to *ViLT*, *ALBEF* is either not as effective as narrowing down the answer candidates or alternatively suffers less from overfitting on the questions. When looking at both results simoultaneously, the latter explanation seems more likely, as *ALBEF* generates its answer freely over all 30522 tokens in its vocabulary.

5.2. RQ2: Coverage

The results of the coverage analysis are shown in table 2. The most notable observation is that for both models, the coverage decreases when using random images, no matter the metric. This is a positive indicator for both models, as it suggests that more internal network states are explored, if more meaningful input information is provided. Another finding is that the edge-case coverage metrics *Boundary*- and *Strong*-coverage suffer more from switching off the image features compared to the main-region metric *K-Section*-coverage. For the behaviour of the network, this could mean that the text input activates most neurons in their main activation part, while the image input is responsible for very low or high activation values. A vertical comparison between *ViLT* and *ALBEF* reveals that both models react similar to the random images. However, there is an even smaller drop in *K-Section*-coverage for [19] after switching off the image input. Following the previous remarks to interpret this finding, [19] tends to use the text input even more to activate its neurons in their main-regions. Besides, it becomes apparent that the *Tanh* activation function is suboptimal when it comes to measuring coverage. Due to its output being bounded between -1 and 1, the maximum size for one

interval in *K-Section*-coverage is $\frac{2}{K}$, whereas for ReLU or GeLU activations this interval can be infinitely large. Thus, it is more likely for neurons to cover each section resulting in a high *K-Section*-coverage. Additionally, if for a neuron n_i the reference activations n^i_{min} and n^i_{max} are close to -1 and 1, respectively, it becomes increasingly harder to find activation values that are lower or higher than the corresponding reference value. As a consequence, *Boundary*- and *Strong*-coverage are expected to be relatively small. The first explanation is mostly confirmed by the results, as the *K-Section*-coverage for the *Tanh* activation is very high for both experiments. Nevertheless, it comes as a surprise that the *Tanh* edge-case coverages for the normal images are on the same level as for the *GeLU* activation.

5.3. RQ3: Adversarial text

Lastly, the results from the adversarial experiments can be seen in table 3. For any of the experiments, almost all of the 30 queue attempts were used up during the attack. It can be concluded that both models are susceptible to adversarial examples. An unexpected observation is that the success rate for both models is higher with the normal images compared to the random images. Initially, one might think that the models predict their top answers in the random settings with less certainty and as such they can be fooled more easily. However, the opposite seems to be the case. The underlying reason might be that for those examples where the model was able to predict the correct answer without image information, there are fewer reasonable answer alternatives. This would make it harder for the attacker to fool the network. Another observation is that attacks on *ALBEF* are more successful. Since in general, seq2seq models tend to be more robust against adversarial examples [8], this might be due to the attack strategy being superior to the one used on *ViLT*. It remains subject to future research to test out more attack strategies to confirm or modify this theory.

6. Conclusion

In conclusion, this research paper has presented and compared two multimodal vision-language models, *ViLT* [16] and *ALBEF* [19], in their ability to perform visual question answering tasks. Through a thorough analysis of their accuracy, coverage, and adversarial robustness, this work has provided insights into the behavior of these VQA models and the role of the text input in their performance. Furthermore, this study has revealed differences between the two models in terms of their architectural approaches. Overall, this work contributes to the understanding of multimodal vision-language models and their potential applications in various contexts.

Model	normal img.	random img.	change
ViLT	0.84	**0.47**	↓ 44.05%
ALBEF	**0.85**	0.39	↓ **54.12%**

Table 1. **Accuracy.** Weighted as described in 3 with full and with random images as input for both models. The third column describes how much the accuracy dropped when using random images.

		normal img.		random img.	
Model	Metric	Tanh	GeLU	Tanh	GeLU
ViLT	K-Section	**99.95%**	90.18%	**98.10%**	75.71%
	Boundary	34.18%	30.89%	5.08%	17.25%
	Strong	32.55%	34.90%	5.73%	13.80%
ALBEF	K-Section	-	**97.80%**	-	**92.40%**
	Boundary	-	33.85%	-	12.67%
	Strong	-	36.98%	-	16.28%

Table 2. **Coverage.** *K-Section-, Boundary- and Strong-Coverage* as defined in [22]. For both models, the values are calculated with normal and with random images. As shown in figs. 1a and 1b, the coverage for *ViLT* [16] is calculated after two different activations, namely *Tanh* and a *GeLU*. For *ALBEF* [19], the coverage is computed after one *GeLU* activation.

Model	Image	Success rate	Avg. num queries
ViLT	normal img.	35.0%	27.8
	random img.	25.0%	28.91
ALBEF	normal img.	**53.75%**	**30.0**
	random img.	38.75%	**30.0**

Table 3. **Adversarial text.** In total, 80 adversarial text attacks were executed for each combination of model and input. If an example was already misclassified by the network, it was skipped and not counted towards the number of attacks. The maximum number of queries was set to 30.

References

[1] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In *European Conference on Computer Vision*, pages 422–440. Springer, 2020. 1

[2] Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4971–4980, 2018. 2

[3] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–

2433, 2015. 1, 2

[4] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. In *European Conference on Computer Vision*, pages 202–221. Springer, 2020. 1

[5] Dave Zhenyu Chen, Qirui Wu, Matthias Nießner, and Angel X. Chang. D3net: A speaker-listener architecture for semi-supervised dense captioning and visual grounding in RGB-D scans. *CoRR*, abs/2112.01551, 2021. 1

[6] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. UNITER: learning universal image-text representations. *CoRR*, abs/1909.11740, 2019. 1

[7] Zhenyu Chen, Ali Gholami, Matthias Nießner, and Angel X Chang. Scan2cap: Context-aware dense captioning in rgb-d scans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3193–3203, 2021. 1

[8] Minhao Cheng, Jinfeng Yi, Pin-Yu Chen, Huan Zhang, and Cho-Jui Hsieh. Seq2sick: Evaluating the robustness of sequence-to-sequence models with adversarial examples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3601–3608, 2020. 4, 5

[9] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. *arXiv preprint arXiv:1606.01847*, 2016. 1, 2

[10] Siddhant Garg and Goutham Ramakrishnan. Bae: Bert-based adversarial examples for text classification. *arXiv preprint arXiv:2004.01970*, 2020. 2

[11] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 1, 2

[12] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017. 2

[13] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015. 3

[14] Guy Katz, Clark Barrett, David L Dill, Kyle Julian, and Mykel J Kochenderfer. Reluplex: An efficient smt solver for verifying deep neural networks. In *International conference on computer aided verification*, pages 97–117. Springer, 2017. 1

[15] Jaekyum Kim, Junho Koh, Yecheol Kim, Jaehyung Choi, Youngbae Hwang, and Jun Won Choi. Robust deep multi-modal learning based on gated information fusion network, 2018. 1, 2

[16] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, pages 5583–5594. PMLR, 2021. 2, 3, 5

[17] Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. Chapman and Hall/CRC, 2018. 1, 2

[18] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021. 2

[19] Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Deepak Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*, 2021. 2, 3, 4, 5

[20] Wei Li, Can Gao, Guocheng Niu, Xinyan Xiao, Hao Liu, Jiachen Liu, Hua Wu, and Haifeng Wang. UNIMO: towards unified-modal understanding and generation via cross-modal contrastive learning. *CoRR*, abs/2012.15409, 2020. 1

[21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2

[22] Lei Ma, Felix Juefei-Xu, Fuyuan Zhang, Jiyuan Sun, Minhui Xue, Bo Li, Chunyang Chen, Ting Su, Li Li, Yang Liu, et al. Deepgauge: Multi-granularity testing criteria for deep learning systems. In *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*, pages 120–131, 2018. 1, 2, 4, 5

[23] John X Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. *arXiv preprint arXiv:2005.05909*, 2020. 4

[24] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. Deepxplore: Automated whitebox testing of deep learning systems. In *proceedings of the 26th Symposium on Operating Systems Principles*, pages 1–18, 2017. 1, 2, 4

[25] Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 1085–1097, 2019. 4

[26] Junha Roh, Karthik Desingh, Ali Farhadi, and Dieter Fox. Languagerefer: Spatial-language model for 3d visual grounding, 2021. 1

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2

[28] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, and Furu Wei. Image as a foreign language: Beit pretraining for all vision and vision-language tasks, 2022. 1, 2

[29] Xinyu Wang, Yuliang Liu, Chunhua Shen, Chun Chet Ng, Canjie Luo, Lianwen Jin, Chee Seng Chan, Anton van den Hengel, and Liangwei Wang. On the general value of evidence, and bilingual scene-text visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10126–10135, 2020. 2

[30] Xuezhi Wang, Haohan Wang, and Diyi Yang. Measure and improve robustness in nlp models: A survey. *arXiv preprint arXiv:2112.08313*, 2021. 1

[31] Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. Visual entailment: A novel task for fine-grained image understanding. *CoRR*, abs/1901.06706, 2019. 1

[32] Xiaofei Xie, Lei Ma, Felix Juefei-Xu, Minhui Xue, Hongxu Chen, Yang Liu, Jianjun Zhao, Bo Li, Jianxiong Yin, and Simon See. Deephunter: a coverage-guided fuzz testing framework for deep neural networks. In *Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis*, pages 146–157, 2019. 1

[33] Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian. Deep modular co-attention networks for visual question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6281–6290, 2019. 2

[34] Wei Emma Zhang, Quan Z Sheng, Ahoud Alhazmi, and Chenliang Li. Adversarial attacks on deep-learning models in natural language processing: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(3):1–41, 2020. 2