

# Adversarial Evaluation of Multimodal Models under Realistic Gray Box Assumptions

Ivan Evtimov  
U. of Washington\*

Russell Howes  
Facebook AI

Brian Dolhansky  
Reddit\*

Hamed Firooz  
Facebook AI

Cristian Canton  
Facebook AI

## Abstract

This work examines the vulnerability of multimodal (image + text) models to adversarial threats similar to those discussed in previous literature on unimodal (image- or text-only) models. We introduce realistic assumptions of partial model knowledge and access, and discuss how these assumptions differ from the standard “black-box”/“white-box” dichotomy common in current literature on adversarial attacks. Working under various levels of these “gray-box” assumptions, we develop new attack methodologies unique to multimodal classification and evaluate them on the Hateful Memes Challenge classification task. We find that attacking multiple modalities yields stronger attacks than unimodal attacks alone (inducing errors in up to 73% of cases), and that the unimodal image attacks on multimodal classifiers we explored were stronger than characterized text augmentation attacks (inducing errors on average in 45% and 30% of cases, respectively).

## 1. Introduction

Multimodal reasoning has become an important ingredient in classification tasks for integrity enforcement for content in ad networks and on social media [17, 37, 39]. Posts with text and images are often evaluated for illegal, harmful, or hateful content with models that process both language and visual information. These use cases are subject to pressure from adversarial bad actors with ideological, financial, or other motivations to circumvent models that identify violating content. For example, a politically extremist group attempting to post misleading or violent content without being detected and removed might attempt to circumvent such models. A seller in an online marketplace may try to obscure a listing for counterfeit goods, drugs, or other banned products.

In this work, we contribute to the understanding of how multimodal models may be vulnerable to threats ob-

\*Work done while at Facebook AI.



Figure 1. Examples of hateful and non-hateful memes in the Hateful Memes dataset [19] and adversarial image and text inputs like the ones we generate. Images originally available [here](#) and accessed on Oct 28th, 2020. Image above is a compilation of assets, including © Getty Images.

served against image-only and text-only models. As a well-representative case study, we focus on the Hateful Memes Challenge and Dataset [19]. This Challenge centers on the binary classification problem of labelling memes (images with embedded text) as “hateful” or “non-hateful”. The notion of “hatefulness” in the Dataset is determined by how the concepts portrayed in the image interact with the meme text as shown in Figure 1.

Two unexplored points need to be addressed to better understand the risk to multimodal classification in practical settings. First, previous adversarial work typically assumes one of two threat models: “white-box” attacks with full access to a target model, and “black-box” attacks allowing only access to model inputs and outputs. In practice, however, shared or inferred knowledge about such classifiers lies somewhere in between these two extremes. Some classifiers may use public, off-the-shelf components (for example, object detector models) in their classification pipeline, while other information about the architecture is unknown. This partial knowledge, while not as helpful to an attacker as full access to model weights, may still allow more pow-

erful attacks than input/output access alone.

Second, it is not obvious how an image-based adversarial attack would impact a multimodal classifier compared to a unimodal, image-only classifier. Would the additional text input improve the robustness of the multimodal classifier, or would the multimodal classifier be *less* robust if both the image and text components can be adversarially attacked?

This paper attempts to address this knowledge gap by quantifying how adversarial examples and text augmentations affect different variants of multimodal classification, under realistic threat models. Our major findings include:

- Adversarial vulnerability does not really depend on the mechanism of combining text and image features (known as multimodal “fusion”); models with different types of fusion fail under adversarial inputs at similar rates.
- Although the multimodal Hateful Memes models are heavily text-dependent, image-only attacks can impact classifier accuracy more than text-only attacks.
- Reusing public off-the-shelf components enables weaker adversaries to carry out successful attacks. In particular, models using pretrained object detectors without fine tuning are vulnerable to attack.
- Under “gray-box” assumptions, combined attacks against both modalities (image and text) are stronger than adversarially perturbing only one modality.

## 2. Related Work

Machine learning has long been evaluated for security and robustness [6, 24, 25]. With the advent and wide deployment of neural networks for image and text processing in numerous practical scenarios, new threats have emerged as well. Since 2013, the research community has focused extensively on *adversarial examples* [36], images that a machine learning model misclassifies, even though they are visually similar (for humans) to benign images the model handles correctly. Subsequent literature has been characterized by an ongoing attack/defense cycle [4, 2, 3, 4, 38]. We apply the algorithm from Madry *et al.* [26] in our work, and discuss its details in Section 3. We also note that some work in this space has explored adversaries without access to the model [14, 28] and we also draw on the lessons from the “transferability” research literature for our gray box attack models [23].

Natural language processing models have seen their share of attacks as well. Many approaches for generating adversarial text from a benign string use a two-step process that is repeated until an adversarial string is found. In stage one, the algorithm selects the most “important” input token to modify; this can be done either with gradient information [5, 9, 20] or with masked queries [1, 11, 16, 21, 29, 42].

In the second step, the selected text is replaced with a suitable candidate; some works simulate typos in words [20], others pick neighboring words in a semantically-aware embedding space [16, 30], and others use a language model (such as BERT [8]) to pick adversarial sentences that read “naturally” [21]. These attacks are standardized and implemented in a popular GitHub repository [27].

A limited set of works has explored the vulnerability of multimodal models but none (to our knowledge) has focused on the classification setting that we study. There exist image-based attacks on VQA that produce answers of the attacker’s choosing [32, 41]. Others have used adversarial modifications in an intermediary feature space in the training process to produce more generalizable models [10] but they have not produced images that generate those features.

## 3. Methodology

Our goal for all examples in this section is to create memes that are either hateful but misclassified as non-hateful, or non-hateful but misclassified as hateful. The memes generated by our attacks must also retain their original meaning, as judged by human users. We use adversarial examples in the image domain and adversarial text augmentations for the text domain. We provide background on multimodal classification in Appendix A and further specify the different threat models that we evaluate in Appendix B.

### 3.1. Attacks in the Image Domain

In all cases, we adopt the projected gradient descent (PGD) algorithm from [26] to modify the image while holding the text constant. We provide full details of our attack implementations along with hyperparameter and model choices in Appendix C. Here, we summarize the 4 major types of attacks we consider.

**Full-Access and Dataset-Access Attacks** In full-access scenarios, an adversary backpropagates through the exact model under attack while using a binary cross-entropy loss function with adversarial labels (opposite of the ground truth). In the dataset-access case, the adversary cannot access the model they wish to fool. However, they can train their own multimodal model and use it to generate adversarial examples in a full-access fashion. As with previous research on adversarial examples transferability [23], we find that this method yields strong attacks.

**Feature Extractor-Access Attacks** When multimodal classification is based on image region features, adversaries have access to a public, off-the-shelf component used in the prediction (such as the Faster R-CNN object detector [31]). In order to choose adversarial features without using gradient or query information from the multimodal model, adversaries can aim to disrupt interactions between the two modalities. Thus, an adversary seeking to make a hateful meme be classified as non-hateful could add perturbations

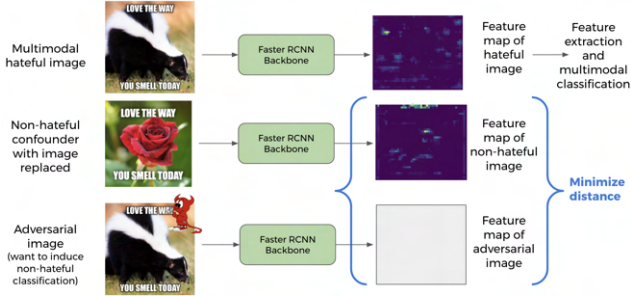


Figure 2. Illustration of one of the “gray box” image-based strategies we explore.

that shift the features that the image of the hateful meme produces to the features of the image from the non-hateful confounder. We visualize this idea in Figure 2.

### No-Access Attacks by Technically Savvy Adversaries

In some scenarios, the adversary does not have access to the weights of any model trained on this task and cannot query them. They can instead carry out a so-called ensemble attack on a set of standard public computer vision classifiers. In this kind of attack, adversarial examples are generated with a white-box attack that averages the gradient from  $n$  models.

### No-Access Attacks by Adversaries with No Expertise

In scenarios with no query access, the adversary cannot use any gradient information (approximated or not) to generate their attacks. They can, however, introduce arbitrary modifications to the image that preserve its message. While there are multiple ways to do this, we use Gaussian noise of the same magnitude as our adversarial noise as a stand-in.

## 3.2. Attacks in the Text Domain

We also perform text-based attacks while maintaining the original image to study the importance of this modality to the adversarial robustness of multimodal classification. We introduce two kinds of text attacks: guided (corresponding to full-access and dataset-access image attacks) and random augmentations (corresponding to no-access scenarios). In both cases, we use the following set of augmentations: inserting emojis, replacing characters with their “fun fonts” equivalent, replacing letters with random other letters or with random unicode characters, inserting typos, splitting words. Full details of our algorithm are given in Appendix D.

## 4. Experimental Results and Observations

We apply each of the methods described in Section 3 to generate adversarial examples and adversarially augmented text. In each case, we start with an example in the Hateful Memes test set and modify it with the goal that the prediction for that example changes to the opposite class from its ground truth. However, recall from Figure 1 that the multi-

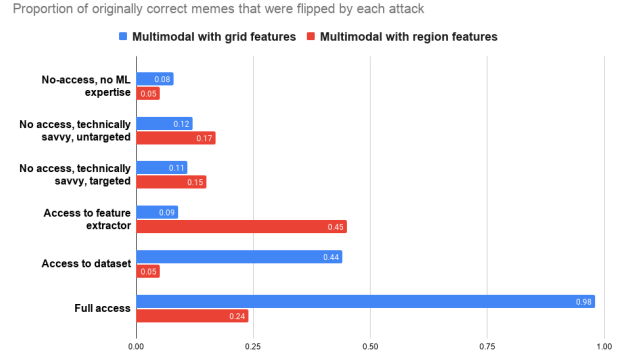


Figure 3. Averaged proportion of memes that were originally correct but misclassified after adversarial modifications of the image. For each of the two categories of multimodal models, we average the metric for each individual model over the three different models in that category.

modal models we work with only achieve 60-70% accuracy and 0.6-0.7 ROC AUC on the test set. In other words, the models misclassify a significant portion of the test set, even without adversarial modifications.

To measure only the effect of our attacks, we report the *proportion of memes that were correctly classified when clean but misclassified when adversarial* out of the memes that were classified correctly to begin with. Formally, for a model  $f$  and a dataset of memes  $\mathcal{D} = \{(x_i, y_i)\}$  where  $x_i$  is a “clean” meme containing text and image and  $y_i$  is its ground truth label, we generate adversarial memes  $x_i^{\text{adv}}$ . Then, we report:

$$\frac{\sum_i \mathbb{1}\{f(x_i) = y_i \text{ and } f(x_i^{\text{adv}}) \neq y_i\}}{\sum_i \mathbb{1}\{f(x_i) = y_i\}}$$

### 4.1. Image Adversarial Examples

For image attacks, we report results by averaging over all models in the two categories of image feature extractors: grid features and region features. Results are given in Figure 3.

To put our attacks in context, first observe the least powerful adversary (inserting Gaussian noise) can flip only around 5 to 8% of memes while the full-access adversary (who backpropagates through the model) can flip 98% of memes that were originally correct.<sup>1</sup> Next, “gray box” attacks with similar effectiveness exist for both grid feature models and region feature models. Dataset-access adversaries can flip 44% of originally correct memes on average and adversaries with access to the region features extractor can achieve 45%. Additionally, technically savvy adversaries with no access can also induce errors in both categories of models at about the same rates (12-17%).

<sup>1</sup>With region feature models, we did not achieve 100%; see App. C.2.

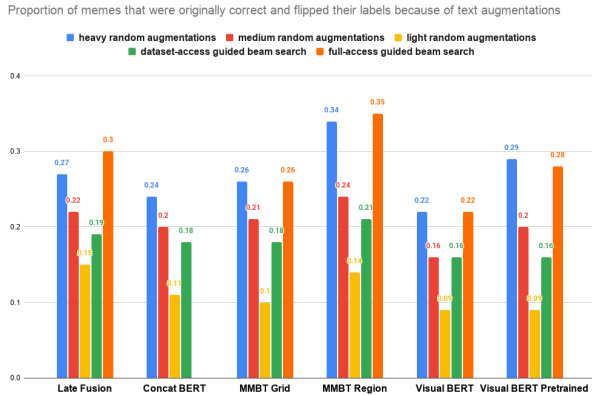


Figure 4. Performance of character-based text augmentations as an adversarial strategy. Each bar represents the proportion of memes that flipped their label after adversarial modifications to the text out of all memes that were classified correctly with no augmentations.

This suggests that the nature of the image feature extraction might not make a model fundamentally more robust.

However, there is an important asymmetry in which attacks perform well on which models. Because grid feature models include retraining the image feature extractor, it takes a strictly more powerful adversary (one with access to the training dataset) to achieve the same goal. This suggests that region feature extractors are, in practice, more at risk of adversarial compromise because an exact component of their operational pipeline is freely available to adversaries.

## 4.2. Text-Only Attacks

We present results on the number of originally correct memes that flipped their label after character-based text augmentations in Figure 4. The first thing to observe about these attacks is that they are less effective than image-based attacks. Attacks that fully disrupt the text (“heavy random augmentations”) without any access to the classifier and attacks with full access to the classifier (“guided beam search”) only cause 25-30% of originally correctly predicted memes to flip their labels. Compare this to the 98% success rate by full-access image adversaries on grid feature models and the 44-45% gray box image adversaries on both sets of models.

The different success rates at different levels of access show that adversarial power is also important here. For example, to control for the risk by an adversary who cannot query the model to an adversary who can query a similar model trained on the same dataset as the target, compare “light random augmentations” and “dataset-access guided beam search.” As a reminder, the maximum edit distance from the original string is the same in both cases, so human understanding of the message is not impeded. However, ad-

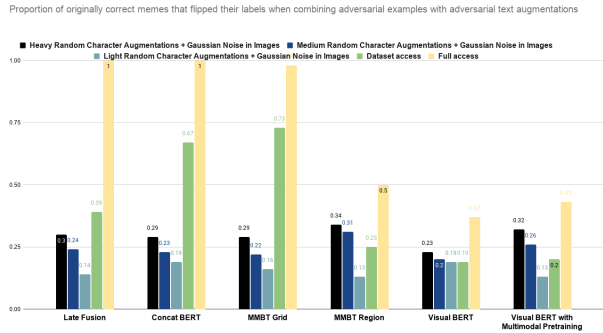


Figure 5. Average proportion of memes that were originally correct but were flipped by attacks that combine image adversarial examples and adversarial text augmentations.

versaries who get to query even a similar model can craft much more powerful attacks – they achieve a 16-19% flip rate, while no-access adversaries only achieve 9-15%.

## 4.3. Combining Text and Image Adversarial Examples

We report results in Figure 5 on attacks that affect both modalities. All attacks are generated by taking the corresponding adversarial image example and adversarially augmented text by an adversary with the same powers. Since there is no equivalent for an adversary who possesses the image feature extractor for the text attacks, we omit this category here. As can be expected, all models perform worse when attacks across modalities are combined. Observe that two models in particular are the worst affected: the concatenation-based “mid fusion” ConcatBERT model and the “early fusion” grid features-based MMBT model. Under gray box assumptions (an adversary with dataset knowledge only), the former is fooled 73% of the time it used to be correct and the latter is fooled 67% of the time. This may suggest that mid and early fusion models relying on grid features are most vulnerable to attacks that are themselves multimodal, even if they do not stand out in vulnerability under attacks in any one single modality.

## 5. Conclusion

This work shows that multimodal models combining text and image data are vulnerable to attacks even when adversaries do not have access to every piece of their pipeline. Strong image-based attacks exist regardless of the feature extractor used. However, it is strictly easier to attack region features-based models as they rely on a publicly available component. Our work opens exciting new avenues for future research. For example, to protect against gray box adversaries, defenses should focus on more robust image feature extraction and aim to reduce transferability of adversarial examples from models trained on the same dataset.



## Acknowledgements

The authors would like to thank the Facebook AI Red Team, Aaron Jaech, Amanpreet Singh, and Vedanuj Goswami for their help. At the University of Washington, Ivan Evtimov is supported in part by the University of Washington Tech Policy Lab, which receives support from: the William and Flora Hewlett Foundation, the John D. and Catherine T. MacArthur Foundation, Microsoft, the Pierre and Pamela Omidyar Fund at the Silicon Valley Community Foundation; he is also supported by the US National Science Foundation (Award 156525).

## References

- [1] Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. Generating natural language adversarial examples. *arXiv preprint arXiv:1804.07998*, 2018. [2](#)
- [2] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *arXiv preprint arXiv:1802.00420*, 2018. [2](#)
- [3] Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 3–14, 2017. [2](#)
- [4] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE, 2017. [2](#)
- [5] 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 *AAAI*, pages 3601–3608, 2020. [2](#)
- [6] Nilesh Dalvi, Pedro Domingos, Sumit Sanghai, and Deepak Verma. Adversarial classification. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 99–108, 2004. [2](#)
- [7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. [8](#)
- [8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. [2](#)
- [9] Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. Hotflip: White-box adversarial examples for text classification. *arXiv preprint arXiv:1712.06751*, 2017. [2](#), [9](#)
- [10] Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-scale adversarial training for vision-and-language representation learning. *arXiv preprint arXiv:2006.06195*, 2020. [2](#)
- [11] Siddhant Garg and Goutham Ramakrishnan. Bae: Bert-based adversarial examples for text classification. *arXiv preprint arXiv:2004.01970*, 2020. [2](#)
- [12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. [6](#), [9](#)
- [13] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. [9](#)
- [14] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. *arXiv preprint arXiv:1804.08598*, 2018. [2](#)
- [15] Huaizu Jiang, Ishan Misra, Marcus Rohrbach, Erik Learned-Miller, and Xinlei Chen. In defense of grid features for visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10267–10276, 2020. [7](#)
- [16] Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. *arXiv*, pages arXiv–1907, 2019. [2](#)
- [17] Kang-Xing Jin. Keeping our platform safe with remote and reduced content review. <https://about.fb.com/news/2020/10/coronavirus/>, March 2020. Online; accessed 29 October 2020. [1](#)
- [18] Douwe Kiela, Suvrat Bhooshan, Hamed Firooz, and Davide Testuggine. Supervised multimodal bitransformers for classifying images and text. *arXiv preprint arXiv:1909.02950*, 2019. [7](#)
- [19] Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. The hateful memes challenge: Detecting hate speech in multimodal memes. *arXiv preprint arXiv:2005.04790*, 2020. [1](#), [6](#), [7](#)
- [20] Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. Textbugger: Generating adversarial text against real-world applications. *arXiv preprint arXiv:1812.05271*, 2018. [2](#)
- [21] Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. Bert-attack: Adversarial attack against bert using bert. *arXiv preprint arXiv:2004.09984*, 2020. [2](#)
- [22] Liumian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. [7](#)
- [23] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770*, 2016. [2](#)
- [24] Daniel Lowd and Christopher Meek. Adversarial learning. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 641–647, 2005. [2](#)
- [25] Daniel Lowd and Christopher Meek. Good word attacks on statistical spam filters. In *CEAS*, volume 2005, 2005. [2](#)
- [26] Aleksander Madry, Aleksandar Makelev, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. [2](#)

[27] 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, 2020. 2

[28] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519, 2017. 2

[29] Danish Pruthi, Bhuwan Dhingra, and Zachary C Lipton. Combating adversarial misspellings with robust word recognition. *arXiv preprint arXiv:1905.11268*, 2019. 2

[30] 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. 2

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 2, 7, 8

[32] Vasu Sharma, Ankita Kalra, Simral Chaudhary Vaibhav, Labhesh Patel, and Louis-Phillippe Morency. Attend and attack: Attention guided adversarial attacks on visual question answering models. In *Proc. Conf. Neural Inf. Process. Syst. Workshop Secur. Mach. Learn*, 2018. 2

[33] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 9

[34] Amanpreet Singh, Vedanuj Goswami, Vivek Natarajan, Yu Jiang, Xinlei Chen, Meet Shah, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. Mmf: A multimodal framework for vision and language research. <https://github.com/facebookresearch/mmf>, 2020. 7

[35] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 9

[36] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 2

[37] The YouTube Team. Protecting our extended workforce and the community. <https://blog.youtube/news-and-events/protecting-our-extended-workforce-and>, March 2020. Online; accessed 29 October 2020. 1

[38] Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *arXiv preprint arXiv:2002.08347*, 2020. 2

[39] Vijaya and Matt Derella. An update on our continuity strategy during covid-19. [https://blog.twitter.com/en\\_us/topics/company/2020/An-update-on-our-continuity-strategy-during-COVID-19.html](https://blog.twitter.com/en_us/topics/company/2020/An-update-on-our-continuity-strategy-during-COVID-19.html), March 2020. Online; accessed 29 October 2020. 1

Model	Image Feature Extraction	Text Preprocessing	Fusion Mechanism	Accuracy	ROC AUC
Late Fusion	Grid features	BERT embeddings	MLP on mean of image and text features	59.8%	0.65
Concat BERT	Grid features	BERT embeddings	MLP on concatenation of image and text features	58.8%	0.66
MMBT Grid	Grid features	Tokenization	Multimodal transformer	62.2%	0.70
MMBT Region	Region features	Tokenization	Multimodal transformer	63.2%	0.70
Visual BERT	Region features	Tokenization	Multimodal transformer	59.9%	0.70
Visual BERT with Multimodal Pretraining	Region features	Tokenization	Multimodal transformer	59.3%	0.68

Table 1. Overview of the multimodal classification models evaluated and the performance metrics we were able to replicate

[40] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017. 9

[41] Xiaojun Xu, Xinyun Chen, Chang Liu, Anna Rohrbach, Trevor Darrell, and Dawn Song. Fooling vision and language models despite localization and attention mechanism. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4951–4961, 2018. 2

[42] Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6066–6080, 2020. 2

## A. Background on Multimodal Classification

There is no one “best” way to achieve good performance for multimodal classification. Here, we describe the most popular approaches, represented by the baseline models in the Hateful Memes challenge [19]. There is generally a 3 step process:

1. Extract features from the image component
2. Process text for classification (In some cases, this includes feature extraction.)
3. Make predictions based on image features and processed text

We summarize the choices for each step in Figure 1 and elaborate in the following two sections.

### A.1. Image Feature Extraction

Step 1 extracts semantic information from images for further processing by other models. There are two common methods for feature extraction – grid-based and region-based. Grid feature extraction uses feature maps from popular convolutional neural networks (CNNs) as image features. For example, for a ResNet [12] network, the output

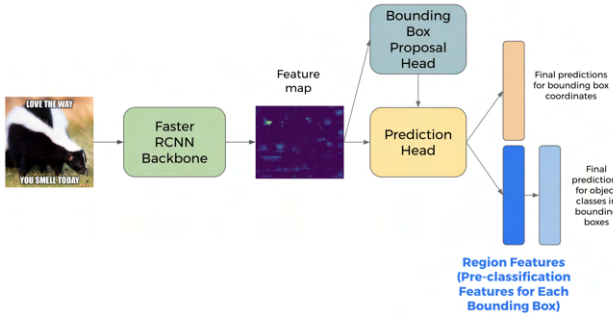


Figure 6. Overview of region feature extraction from images with Faster R-CNN models [31].

of the `res5c` layer is taken as representing the image for downstream multimodal classification.

Region feature extraction is more widely used and is critical to our attack design. The baseline models for the Hateful Memes Challenge use the Faster R-CNN family of models to extract region features. Faster R-CNN processes images in three stages which we illustrate in Figure 6. First, a backbone network produces feature maps distilling the contents of the input image at several different resolutions. Then, a branch of the network (the “proposal head”) uses the feature map to generate rough-estimate proposals for bounding boxes where an object may be located. These proposals are refined by another branch (the “prediction head”) to produce exact coordinates and classes for each proposal, also using the feature map. Finally, non-max suppression is applied so that each object is captured in only one bounding box. The “region features” are the output of the fully connected layer before the softmax classification layer in the prediction head, for each proposed bounding box that survives non-max suppression.

While there is an active debate [15] in the multimodal research community about which features achieve better performance, there is a significant distinction between the two for adversarial analysis. Because grid feature models are more lightweight, multimodal models built on top of them are trained end-to-end. Thus, no off-the-shelf components are used in deployment. By contrast, region feature models are often too heavy to train end-to-end and publicly available, pretrained models are used without fine-tuning. This implies different levels of access for the attacker which we describe in more detail in Section B.

The Late Fusion, ConcatBERT and one version of the MMBT model (developed in [19] and [18]) use grid features while the Visual BERT model (first introduced in [22] and adapted in [19]) and another version of MMBT use region features. All models we work with are implemented in the MMF library [34] and trained on the Hateful Memes dataset [19].

## A.2. Text Preprocessing and Multimodal Fusion

The method used for text preprocessing for the text modality is closely tied to the multimodal fusion mechanism used. It is useful to think of 3 levels of fusion: late, mid, and early. In late and mid fusion, high-level semantic embeddings are extracted from text in a similar fashion to image feature extraction. A unimodal text model is used to pre-process this modality. For example, in the ConcatBERT and Late Fusion models from the Hateful Memes challenge, BERT embeddings are used. But in other cases, such embeddings could be derived by character-based CNNs and RNNs. In those situations, the resulting text embeddings are combined with the image features either through concatenation (mid fusion) or averaging (late fusion) and a “light-weight” multilayer perceptron (MLP) is used on top. In early fusion settings, text tokens are instead fed directly into a multimodal transformer in the same way they are fed to unimodal transformers (such as BERT). To prepare image features for the format required by transformers, affine transformations can be learned (as in MMBT [18]) or engineered to indicate they are image embeddings (as in Visual BERT [22]). In both cases, we treat this preprocessing of the image embeddings as a frozen part of the multimodal transformer that the attacker has no control over – just like the weights and biases deeper in the transformer.

## B. Threat Model

We will consider five different adversaries, each distinguished by their knowledge or technical capability of generating attacks.

To begin with, the most powerful adversary we consider is the classic full-access attacker common to the majority of the adversarial examples research literature. We assume that they possess the architecture and weights of the exact models used in the deployment of multimodal classification. They can, therefore, run and backpropagate through those on their own and adapt their attacks as necessary. Considering this adversary helps define the bounds of what attacks are possible in the worst case for the system designers.

However, attackers with less knowledge than this may also be of concern and we also seek to understand what they are capable of. Consider two important pieces of building multimodal classification: the dataset used to train the models and the image feature extractor used to generate inputs for the image modality.

First, adversaries may possess the dataset used to train the multimodal models even if they do not know which model in particular is being used. Such data sets are often made public for academic research purposes. Adversaries seeking to attack a hatefulness classification system can, therefore, certainly train their own multimodal models to guide their attack generation.

Second, attackers are likely to have access to the exact image feature extractor even if they do not have access to either the exact dataset used or the full model. As we mentioned in Section A, it is a common practice in multimodal classification to use so-called “region features” for images. Those are extracted with publicly available <sup>2</sup> object detectors such as Faster R-CNN [31].

Finally, the system designer applying multimodal classification may choose to not rely either on public datasets or public models for preprocessing. Thus, it is also important to consider adversaries who do not have this level of access. We differentiate between two possible attackers in this category. On the one hand, technically savvy adversaries may use public computer vision models to guide their process of generating adversarial examples. They could, for example, obtain implementations of ImageNet [7] classifiers. On the other hand, adversaries may not have any expertise in machine learning at all. In those cases, they can insert noise or augmentations that are not guided by any model at all.

Thus, the five adversaries we consider are:

1. Full-access attackers possess the multimodal model weights and architecture
2. Dataset-access attackers possess the dataset used to train the multimodal classifiers but not the exact models being used.
3. Feature extractor-access attackers possess the component of the pipeline used to extract image features.
4. No-access, technically savvy attackers have machine learning knowledge but no access to any component of multimodal classification.
5. No-access, low expertise attackers do not have any machine learning knowledge whatsoever.

## C. Further Details on Methodology

### C.1. Formulation of the PGD Objective

The generic adversarial objective for an image  $x$ , a model  $f$ , adversarial loss function  $L$ , and maximum perturbation  $\epsilon$  is as follows:

$$x' = \arg \min_x L(f(x)) \text{ s. t. } |x' - x| \leq \epsilon \quad (1)$$

This objective is solved by gradient descent by using the following update rule:

$$x_{i+1} = \text{Proj}(x_i - \alpha \nabla_{x_i} L)$$

<sup>2</sup>Implementations and weights of those models are available, for example, at <https://github.com/rbgirshick/py-faster-rcnn> and <https://github.com/facebookresearch/grid-feats-vqa>.

Adversarial power	Loss function	Model to backpropagate through	Targets
Full access	Binary crossentropy	Model under attack	Opposite ground truth label
Access to the training dataset	Binary crossentropy	Similar multimodal model	Opposite ground truth label
Access to the image feature extractor	Feature map L2 loss	Public object detector (e.g. Faster RCNN)	Feature map of image from meme of opposite class but with matching text
No access, technically savvy	Feature layer L2 loss	Ensemble of public vision classifiers	Untargeted: away from original Targeted: opposite-class image with matching text

Table 2. Overview of the attack strategies used to instantiate Eq. 1 for image-based attacks

where  $\alpha$  is the learning rate and  $\text{Proj}$  is the projection function on an  $\mathcal{L}_\infty$  sphere of radius  $\epsilon$  around the original image  $x$ . We set  $\epsilon = 0.1$  and  $\alpha = 0.05$  for all experiments that we report results on.

We need to further specify three components for equation 1: the model  $f$  used to generate adversarial examples, the form of the loss  $L$  (e.g. cross entropy or L2 distance), and the targets used in the adversarial examples generation. The choice for each of these varies depending on the threat model and what model the adversary has access to. We summarize our choices for each threat model in Table 2. We summarize our choices for each level of adversarial power in Table 2.

### C.2. Full-Access and Dataset-Access Attacks

Note that for region feature models, we cannot directly use backpropagation to the input  $x$  because object detectors apply non-max suppression to select bounding boxes (and, by extension, the features that become inputs to the multimodal fusion mechanism). Moreover, gradient descent is likely to be unstable as slight perturbations in each step cause disproportionate changes in the bounding box proposals and associated features. Instead, we employ the following two-step procedure for Faster R-CNN features:

1. Generate a single adversarial vector  $y'$  for the multimodal transformer. In this case,  $f$  is the multimodal transformer with the text input fixed and  $L$  is the binary cross-entropy function with adversarial labels.  $y'$  is prepended to the sequence of image feature input vectors while all other vectors are held constant.
2. Generate an adversarial example  $x'$  such that the Faster R-CNN detector produces that vector  $y'$  for all proposed bounding boxes. In this case,  $f$  is the Faster R-CNN classification head with the classification layer removed and  $L = \sum_i \|f(x')_i - y'\|_2$  for each feature vector  $f(\cdot)_i$  output by the Faster R-CNN classification head.

We note that stronger adversaries in the full-access scenario are likely possible. Our exploration only focused



on adversaries that first generate a single adversarial feature to be included in those produced by the region feature extractor and then produce adversarial images that output that feature. This is because we cannot backpropagate end-to-end with region feature extractors (they include a non-differentiable non-max suppression step). While stronger adversaries with full access were outside the scope of this work, those are likely to exist, so the 24% number should be treated as a lower bound.

### C.3. Feature Extractor-Access Attacks

To use their access to an off-the-shelf feature extractor (used in multimodal classification) to carry out an attack, adversaries can instantiate Eq. 1 with two pieces: choosing what features the adversarial images should generate and designing an appropriate loss function.

Recall that the Hateful Memes Challenge Dataset contains non-hateful confounders that were created by hand by replacing images of hateful memes with such that make the overall message of the meme non-hateful even while leaving the text unchanged. (See Figure 1.)

To produce those features, we propose an adversarial loss function that targets the feature map in the detector pipeline that is used to produce both proposal boxes and features. Recall from Appendix A that the region features produced for a given image are computed from the feature map layer of a Faster R-CNN object detector. Thus, if an adversarially modified hateful image produces a feature map corresponding to a non-hateful image, the region features computed will correspond to the non-hateful image. Therefore, we design our loss function so that it penalizes the distance between the feature map of a desired non-hateful target image and the adversarial one we are optimizing.

Formally, let  $x$  be the image of a hateful meme and let  $y$  be the image of a meme with the same text but an image that makes it benign. Further let  $f$  be the Faster R-CNN feature map layer (e.g., c4). Then, in Eq. 1,  $L = \|f(x) - f(y)\|_2$ . The same loss function holds if  $x$  is non-hateful and  $y$  is its hateful counterpart with matching text.

In the Hateful Memes test set, we found 483 memes that had a corresponding counterpart with matching text but the opposite ground truth label. Experimental results with models with region features and feature-extractor access, therefore, report the success rate as a proportion of 483.

### C.4. No-Access Attacks by Technically Savvy Adversaries

Adversarial examples are generated with a white-box attack that averages the gradient from  $n$  models. For models  $f_1, \dots, f_n$ , we modify the PGD objective as follows:

$$x' = \arg \min_x \frac{1}{n} \sum_{i=1}^n L(f_i(x)) \text{ s. t. } |x' - x| \leq \epsilon \quad (2)$$

In all cases,  $f_i$  is taken to mean the output of the final convolutional feature map in the corresponding network (e.g. `res5c` in ResNet). We work with ResNet-152 [12], ResNext-50 [40], Inception-v3 [35], VGG-16 [33], and DenseNet [13]. We further introduce two versions of Equation 2:

- In *untargeted* attacks for original image  $x$  and adversarial images  $x'$ , we set  $L = -\|f(x) - f(x')\|_2$  to create adversarial images that shift the feature map as far away from its original as possible.
- In *targeted* attacks, we set  $L = \|z - f(x')\|_2$  for some target feature map  $z$ . Feature maps are selected as in Section C.3.

## D. Details of Algorithm for Text-Based Adversarial Augmentations

### D.1. Guided Adversarial Text Augmentations

Input to text models such as BERT is discrete, so using gradient-based approaches is not possible. However, we can still use queries to the model to guide a search for adversarial text augmentations. Therefore, we adapt beam search techniques (such as those used in [9]), to the multimodal scenario.

Our adversarial search algorithm works as follows. For a given string we want to adversarially augment, we apply a set of character-based augmentations. At each step of the beam search, we generate multiple different variants at random and only a minimal character augmentation is applied to each one (such as only replacing one letter). We discard augmented strings whose edit distance from the original string exceeds a threshold  $\tau$ . What remains is the “candidate” set. We then rank each member of the candidate set according to how big of a drop in confidence on the correct class for the meme under attack it causes. If the drop by any candidate is big enough to cause an error in classification, we stop there and return the successful augmented string. If none of the candidates causes a classification mistake, we select the top  $k$  to remain in the beam. Then, we repeat this process by generating multiple randomly augmented candidates for every string in the beam, selecting those under the edit distance threshold, and picking the top  $k$  for the next iteration of the beam search.

In white-box scenarios, we use the model under attack to rank the candidates. In light gray-box scenarios, we use any other multimodal model to perform the ranking.

For heavy augmentations, we cap the edit distance at  $\tau = 0.5$ ; for medium augmentations, we cap the edit distance at  $\tau = 0.2$ ; and for light ones, we set  $\tau = 0.07$ .

## D.2. Random Adversarial Text Augmentations

Just as in the image domain, it is useful to generate adversarial text that does not require queries or other access to the model under attack. We use the same set of character-based augmentations as in the guided scenario, but perform random search to pick them instead of a guided beam search. We study 3 levels of adversarial modifications:

- Light: These are selected so that the maximum edit distance from the original matches that of the adversarial strings generated by the beam search.
- Medium: These are selected so that the average edit distance from the original string matches that of the beam search.
- Heavy: We do not restrict the edit distance and allow for maximum corruption of the text string. The text in this case carries no human-interpretable message and violates our attack requirement but this is a useful “upper bound” on the effectiveness of text-based attacks on multimodal classification.