Multimodal Classification for Hate and Sarcasm Detection

Aishwarya Rajasekaran arajasekaran
Dhruti Patel dhrutibenpat
Kriti Faujdar kfaujdar
Shivam Utreja sutreja

1 Introduction

With the world connecting all over social media platforms, expressing emotions through multimodal means has increasingly dominated billions of lives. Hate and disinformation on these platforms is also increasingly multimodal. Some of the most toxic or harmful user content comes in the form of images and videos, accompanied with text.

Memes form a large chunk of traffic on social media platforms. While most of these are harmless and funny in nature, malicious users can use clever combinations of independently innocent images and text to form what may be called "hateful memes" - memes that denigrate people based on qualities such as religion, ethnicity, gender, political alignment etc. As a naive example, a harmless picture of a desert could be combined with the harmless text "Look how many people love you"; making the sentiment of the meme "hateful" as a whole. Automatic detection and removal of such content from social media is a top concern for firms like Facebook and Twitter.

Memes often require a bit of parsing and thinking to get at their meaning, even for reasonably savvy humans, indicating that it's a much harder problem for AI algorithms. They're often highly contextual, referential, ironic, and nuanced. They can also be cryptic and encoded so that only members of some specific enclave of internet users can parse their meaning.

On similar lines, sarcasm is no longer a purely linguistic phenomenon. To detect sarcasm, context needs to be taken from both text and image. With the rapid growth of social media usage, multimodal sarcastic tweets are very popular. In such tweets, there is a small text and contrast is shown using an image. For example: The tweet "Perfect flying weather for April" seems normal but when the image of downpour outside the airplane window is added it becomes sarcastic.

Despite the recent breakthroughs in NLP with BERT and related language models, detecting hate and sarcasm in memes/posts continues to pose a challenge due to its inherent multimodal nature. The model needs to be able to make a combined inference on the visual and textual components of the input in order to correctly identify the underlying sentiment.

2 Problem Statement

We intend to scout for a better model to solve such challenges. The project aims to use multimodal methods to detect contrast in different modalities along with finding relevant associations to be able to achieve better performance. This project will help to detect harmful content that affects the community and society at large which further can be leveraged at different social media platforms to build a better community environment.

3 What you proposed vs. what you accomplished

The list of things proposed in the project proposal, along with their final outcome have been listed below.

1. Acquire and explore Hateful Memes Challenge dataset by Facebook AI: *Completed. In addition, we also spent some time exploring the Sarcasm dataset.*

2. Implement Unimodal baseline model to fetch contexual information from text.:*Completed. Implemented and ran both, text and image unimodal baselines, as well as simple multimodal baselines (with and without finetuning in each case).*

3. Understand the D & R Net architecture for Multimodal model implementation.:*Complete.*

4. Implement the D & R Net model, train and test it for sarcasm detection.: *Incomplete. The implementation of this model depends on extracting ANPs (Adjective-Noun pairs) from an external black-box, which we were unable to get working.*

5. Fine tune the Multimodal model for Hateful Memes dataset.: *Completed. We experimented with intermediate finetuning on several SOTA multimodal architectures from the Facebook MMF model zoo.*

6. Change model components to improve the performance for Hate detection.: *Incomplete. We intended to experiment with changing the multimodal backbone to LXMERT but were unable to get the required Faster RCNN image features.*

7. Analyze the model behaviour and comparative analysis of implemented models.:*Completed.*

8. More fine tuning if required and collect data to present in report.:*Completed.*

9. Work on final report.:*Completed.*

4 Related work

When exploring the unimodal classification baseline, we work with the state of the art architectures for the text and visual modalities. For text-only classification, the BERT architecture (Devlin et al., 2018) has been proven to give exceptional performance. For image classifications there are many architectures that perform well. Even though these have been shown to achieve high classification accuracy in their respective domains, they fall significantly behind human performance in the multimodal scenario as shown in (Kiela et al., 2020). This is simply because in a multimodal scenario, just the image or the text in itself doesn't contain the complete information.

There also exist several multi-modal methods that work much better in such scenarios. One class of these methods is pretty straight-forward and can be broadly classified as *late fusion* methods. These methods essentially use the idea of transfer learning to combine the above mentioned pretrained unimodal features (for example, by concatenation), and fine-tune them for the task at hand.

Apart from these, there exist more recent, sophisticated multi-modal architectures such as ViLBERT (Lu et al., 2019) and VisualBERT (Li et al., 2019) that combine information from the two modalities at a lower level. ViLBERT (Lu et al., 2019) extends the popular BERT architecture to a multi-modal two-stream model, processing both visual and textual inputs in separate streams that interact through co-attentional transformer layers. These multi-modals models extract set of features from visual and text and their associations on various dimensions. While, VisualBERT (Li et al., 2019) aims for extracting rich semantics from image and it's associated text by integrating BERT (Devlin et al., 2018) and pretrained object proposals systems such as Faster-RCNN (Ren et al., 2015). The extracted image features which will be served as inputs tokens to model along with text. They are jointly processed by transformer layers and thus helps to find intricate associations of visuals and text. (Li et al., 2020).

Another such architecture is MMBT(Kiela et al., 2019). It is supervised bi-directional transformer with unimodally pretrained text and image encoder. Unimoadlly pretrained text encoder (BERT) and image encoder (ResNet) are jointly fine-tuned by projecting image embeddings to text token space.

All the above models are available in MMF framework released by Facebook AI Research (Singh et al., 2020). MMF is a modular framework built in pytorch for vision and language multimodal research. It is packaged with reference implementations of state-of-the-art vision and language models . It allows distributed training and is un-opinionated, scalable and fast. MMF also provides a codebase for challenges around vision and language datasets including hateful memes challenge.

5 Datasets

For this project we will be primarily using two data sets: The Hateful Memes Dataset(HMD) provided by Facabook AI (Kiela et al., 2020) and the Sarcasm Dataset(SD) which is publicly available, and constructed by (Yitao Cai and Wan, 2019).

Figure 1: These images depict the Hateful memes and their confounders. Multimodal hateful memes (left), benign image confounders (middle) and benign text confounders (right).

5.1 Hateful Memes Dataset:

The features in this data set are the meme images themselves(with the text overlay still present) and strings of the text from the meme image already extracted. We did not have to extract the text from the meme, as OCR is a solved problem. The HMD is already annotated by group of human annotators who have labelled the memes as either hateful(label=1) or non-hateful(label=0). This labelling was based on Facebook's community guidelines. For memes found to be hateful, benign confounders were constructed, by replacing the image and/or the text of the hateful meme with an alternative that flips the label from hateful back to not-hateful. After filtering out low quality memes and those in violation of Facebook's terms of service (depicting gore/other explicit content), the present dataset consists of 10,000 memes for phase I of the competition.The competition website also released an additional 540 validation images, and 2000 test images in the second stage of the competition.

There are five different types of memes: (1) multimodal hate, (2) unimodal hate where one of the two modalities were already hateful on their own, (3) benign image and (4) benign text confounders and finally (5) random not-hateful examples.The dataset comprises of memes in the following percentages: 40% multimodal hate, 10% unimodal hate, 20% benign text confounder, 20% benign image confounder, 10% random non-hateful.

Since this is an ongoing challenge, the labels for neither of the test splits were available. For the purpose of this project, we used dev_seen for validation and dev_unseen as our test split.

Need of confounders: There could be an inherent bias in the type of imagery and the kind of vocabulary used to convey hate in a dataset collected off the web. To remove this and to truly test the model's multimodal capacity, the dataset includes "confounder" memes that would have the opposite effect to the original. Originally mean-spirited meme are turned into something appreciative or complementary.

Exceptions: Its important to note the exceptions to Facebook's definition of hate - attacks against individuals/celebrities is not considered hate if the attack is not based on any of the protected characteristics listed in the definition (gender, caste, race etc.). Attacks on groups perpetrating hate (like terrorist organizations) is not considered hateful either.
Statistics of the Hateful memes dataset are given in table 1

5.2 Sarcasm Dataset:

Each sample in this data set is an image-text pair. The data set is collected from Twitter by querying special hashtag(e.g. #sarcasm, #sarcastic, #irony, #ironic etc.) for positive samples (i.e. sarcasm) and the others without such hashtags as negative samples (i.e.non-sarcasm). There is no bias removal done in the original dataset. The dataset has been divided into training set (80%), development set (10%) and test set (10%).
Details are given in Table 2

5.3 Data preprocessing

Hateful Memes Data: For the baseline models, we preprocess the data by resizing the image to 224*224 and balancing the samples in each class

	Train	Dev_seen	Dev_unseen	Test_seen	Test_unseen
Non-Hateful	5481	253	340	*Unknown*	*Unknown*
Hateful	3019	247	200	*Unknown*	*Unknown*
All	8500	500	540	1000	2000

Table 1: Statistics for the Hateful Memes Dataset.

soft boiled egg swept in bacon

my transparent kongos just came from asmara . the streets ain 't ready for this heat

well , we can thank our lucky stars this thing is still standing .

Just when i thought my day couldn't get any better ...

Figure 2: Examples from the Sarcasm dataset (images with their accompanying text). The two images on the left have label=0 (non-sarcastic), while the two on the right have label=1 (Sarcastic)

	Train	Validation	Test
Sarcasm	8642	959	959
Non-Sarcasm	11174	1451	1450
All	19816	2410	2409

Table 2: Statistics of Sarcasm Dataset

in the training split. This is done within the training code pipeline itself.

Sarcasm Data: To remove obvious text biases, we remove those datapoints that have mentions of terms like "sarcastic", "joke", "humor", "irony" etc., within the text of the tweet itself. See the *converter.py* script for an exhaustive list.

Since, the both these data sets were tested on common models and frameworks, we had to convert their data files to same format.

6 Baselines

The following baselines were implemented by us independently in *Pytorch*, using the pretrained models from the *torchvision* and *transformers* modules.

6.1 Unimodal Baselines

Unimodal methods require only one modality from the data. In our case, it can be either be text or image. For image data we used pretrained RESNET-152 and extracted 2048-dimensional features. For the text data, we used pretrained BERT-base-cased to extract the 748 dimensional feature vector corresponding to the [CLS] token. We experimented with and without fine tuning on these base models. In case of fine tuning, we train over the pretrained weights across all layers in the model, whereas for without fine tuning, we freeze all the layers before the feature extraction layer.

	Val Accuracy	Test Accuracy
Text only	57.2%	57.78%
Image Only	53.8%	52.78%

Table 3: Hateful memes dataset: Results for unimodal classification, **without finetuning**.

	Val Accuracy	Test Accuracy
Text only	79.75%	79.99%
Image Only	64.73%	63.88%

Table 4: Sarcasm dataset: Results for unimodal classification, **without finetuning**.

For Fine-tuning unimodal for text we add a hidden layer of size 300 over the 768 dimensional feature vector and for image we add a hidden layer of size 1000 over the 2048 dimensional image

feature vector. Similarly for multimodal we add a hidden layer of size 1000 over the concatenated features of 2816 dimension.

For the without fine-tuning cases, the results were obtained by training an MLP Classifer with 2 hidden layers of sizes 400 and 25 respectively, $\alpha = 10^{-5}$ and max_iter = 700. This set of hyperparameters was found using grid search for the hidden layer sizes and the value of α. The grid search was only done for the multimodal case, and then the classifier network was kept the same for all other baselines.

	Val Accuracy	Test Accuracy
Text only	54.6%	55.3%
Image Only	52.1%	53%

Table 5: Results for unimodal classification on hateful memes, **with finetuning**.

6.2 Multimodal Baselines

Multimodal method requires both images and texts. We take the image data and pass it through the vision model and extract the last layer feature representation. Similarly, extract the text feature representations from the [CLS] token in the last layer. We then concatenate them together and add fully connected layers on top. We tried the above architecture both, with and without fine tuning on the text and vision modules.

Results of unimodal classification on hateful memes with fine-tuning and without finetuning are presented in Table 3 and Table 5 respectively.Similarly Table 4 summarizes results for sarcasm data set. Results of multimodal late fusion for both the datasets are presented in Table 6 and Table 7

6.3 Inferences from baselines

Our inferences from the above experiments are described below.

- For hateful memes, multimodal model without fine-tuning gives the best result. However, these were also very poor (sub-60% accuracy on binary class labels).
- Unimodal models perform even worse, due to confounders added in the dataset. This highlights both, the importance of removing biases when building datasets for a multimodal task; as well as the fact that addition of confounders makes this problem very challenging.
- For the sarcasm dataset, the performance of unimodel text and multimodal are comparable. This highlights that the sarcasm dataset is heavily ridden with text bias, and the preprocessing done isn't enough to remove it. We would expect any reasonable multimodal method to give more than 80% accuracy on this dataset.

	Val Accuracy	Test Accuracy
Without Fine-tuning	58.2%	58.9%
With Fine-tuning	55.6%	54.8%

Table 6: Hateful Memes: Results for multimodal late fusion

	Val Accuracy	Test Accuracy
Without Fine-tuning	80.25%	80.37%

Table 7: Sarcasm dataset: Results for multimodal late fusion.

7 Your approach

In addition to implementing unimodal and multimodal baselines, we took advantage of architectures available in Facebook's Multi Modal Framework (MMF). We performed finetuning and intermediate finetuning experiments on Hateful memes dataset and sarcasm datasets using MMBT, VilBERT and Visual BERT models. Implementation details are presented in Table 8. We first describe the underlying multimodal architechtures we will use throughout our experiments.

7.1 Architectures

- **MMBT**
  It takes text and image encoder which are pretrained on unimodal tasks, combines them and jointly fine-tunes them. It provides the flexibility to replace the text and image encoders with better alternatives. Image encoder is output of pooling layer of ResNet-152 but instead of pooling generating only single vector, N separate vectors are being generated. Pooling is done over grids in the image. Pretrained BERT is being used

as a text encoder. Both image and text encodings are assigned segment ids and passed to transformer along with positional embeddings. Idea is to employ self-attention over both the modalities at the same time.

- **ViLBERT**
  In BERT, we use a multi-headed attention block which computes three vectors, Q,K,V for hidden layers in the encoder section. We here modify the query conditioned key-value attention mechanism to include text(BERT) as well as visual(Faster RCNN model) modality for our model. We extend the architecture to a multi-modal two-stream model, processing both visual and textual inputs in separate streams that interact through co-attentional transformer layers. Forcing the pretrained weights to accommodate the large set of additional visual 'tokens' may damage the learned BERT language model. Instead, we develop a two-stream architecture modelling each modality separately and then fusing them through a small set of attention-based interactions. This approach allows for variable network depth for each modality and enables cross-modal connections at different depths. (Lu et al., 2019)

- **Visual BERT**
  VisualBERT consists of a stack of Transformer layers that implicitly align elements of an input text and regions in an associated input image with self-attention (Li et al., 2019). The image region proposal are obtained using a faster RCNN model. These features help the model to learn the relation between image region and text phrases which in turn help to extract the tone of information using two modalities.

7.2 Fine-tuning on individual datasets

Hateful memes dataset: We finetuned MMBT-Grid, VilBERT and Visual BERT models from MMF framework on Hateful Memes dataset. Results achieved are reported in table 9. VilBERT and Visual BERT models use Faster RCNN features which were made available by MMF for this particular dataset.

Sarcasm dataset: We finetuned MMBT-Grid and Late-Fusion models from MMF framework on Sarcasm dataset. Results achieved are reported in table 10. Original plan was to finetune ViLBERT and Visual BERT models too on Sarcasm dataset, but these models need Faster RCNN features which we were unable to extract.

7.3 Intermediate finetuning on Sarcasm followed by finetuning on Hateful Memes dataset

Since the size of Hateful meme dataset was small, intermediate transfer learning seemed an efficient way to increase the performance of models, as described in (Pruksachatkun et al., 2020). We move forward performing intermediate training on sarcasm dataset and then fine-tuning on Hateful Meme Dataset. The reason of performing intermediate training on Sarcasm dataset are:

- **Similarity between downstream task and intermediate task:** Sarcasm detection also works on the concept of using both text and image modalities together, to perform a binary classification. Additionally, on exploring the datasets; conveying hate through cryptic memes was observed to be highly correlated to a sarcastic tone. Thus it seemed encouraging to use sarcasm as an intermediate task.

7.4 Results

The quantitative analysis for experiments on Hateful Memes dataset are summarized in table 9, on Sarcasm dataset are summarized in 10 and for intermediate finetuning experiments are summarized in 11.

8 Error analysis

We manually went through the wrongly predicted datapoints for our best performing models, for both the datasets. We observed the following general trends in the type of cases where the trained model was consistently making mistakes.

8.1 Sarcasm Dataset

Poor data collection technique.
On manual analysis of dataset, we figured that labelling in the sarcasm dataset is not as thorough as hateful memes dataset. There are some examples which clearly looks to be sarcastic but are not labeled sarcastic. For example: Text associated with following image is "oh

Table 8: Implementation details and Hyperparameters

Model	Batch size	LR	Text Encoder	Image Encoder
Concat BERT	16	1e-5	BERT	ResNet-152
MMBT-Grid	16	1e-5	BERT	ResNet-152
ViLBERT	16	1e-5	BERT	FasterRCNN
Visual BERT	16	5e-5	BERT	FasterRCNN

Table 9: Models Accuracy - Hateful Meme Dataset

Model	Val Acc	Test Acc	Test(AUC-ROC)
MMBT-Grid	61.00	66.11	65.75
ViLBERT	62.40	67.41	70.42
Visual BERT	50.60	62.59	61.16

yeah , attack a ten year old child . classy .”

Poor data collection technique
“oh yeah , attack a ten year old child. classy .”
True label: 0 Predicted label: 1

The reason for this could mainly be the way the data has been collected. The creators only considered those examples to be sarcastic which were marked with certain hashtags such as #sarcasm, #sarcastic, #irony etc and the examples without such hashtags were marked non-sarcastic. The above has no such tag , so it was labeled as non-sarcastic. But if look at the text associated with the example ”oh yeah , attack a ten year old child . classy .”, the tone is clearly sarcastic.

High dependence on contrast.

Sarcasm detection seems to heavily rely on cross modal contrast to perform the task efficiently. But not all tweets uses contrast in their text and image to convey sarcasm. The paper (Castro et al., 2019) presents few examples supporting the fact of cross-modal contrast in sarcastic tweets but this is not always true. The above can be validated by few examples observed while performing error analysis. For example, the tweet which consists the text : *“dump your furniture on the sidewalk & leave . the city of # guelph will clean it up . # guelph # students # university of # guelph”* and the image containing the furniture left on the sidewalk, the model predicted it as non-sarcastic but in real it was sarcasm.

High dependence on contrast
“dump your furniture onthe sidewalk leave . the city of guelph willclean it up . # guelph # students # university of# guelph”
True label: 1 Predicted label: 0

Additional text in the image.

During our error analysis we encountered examples where there was text in the image that was indicating sarcasm. Our models are not able to capture this text because we are not scraping the text off image and considering

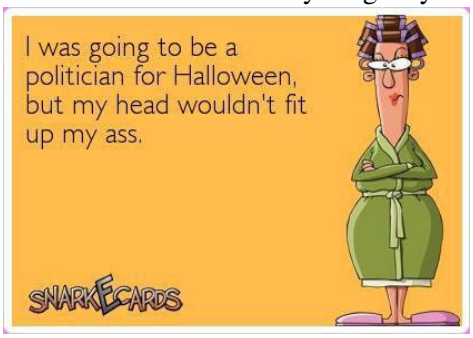
Table 10: Models Accuracy - Sarcasm Dataset

Model	Val Acc	Test Acc	Test(AUC-ROC)
MMBT-Grid	87.47	88.25	95.37
Late Fusion	87.43	86.51	94.35

Table 11: Models Accuracy - Intermediate experiments

Model	Val Acc	Test Acc	Test (AUC-ROC)
MMBT	60.20	61.60	68.34

it for predictions. In such cases, predictions are mainly based on the text associated with image in dataset. For example: below, the text associated with example is "halloween" which doesn't indicate anything by itself.



Additional text in the image
"halloween"
True label: 1 Predicted label: 0

To find the intended sarcasm, text which is part of the image needs to be looked at and our models are not considering that. Hence the model is predicting the image as non-sarcastic. This could be improved by adding text from image to be part of dataset as well.

8.2 Hateful Memes Dataset

Difficulty in handling confounders.
The hateful memes dataset has a subset of non-hateful memes which were created by using originally hateful memes and switching out either their text, or their image, to make the final result non-hateful. This was done to force the models to rely on cross-modal reasonings to make final conclusions. However, as we observed during our error analysis, a lot of the mistakes made by even the best models were in this category of non-hateful memes. This suggests that even the deep multimodal architectures considered are unable to let go of their biases from pre-training, even after fine-tuning.

Difficulty in handling confounders
"Pakistanis in every Indian page"
True label: 0 Predicted label: 1

Difficulty in multi-image format.
In the examples where multiple images have been merged to create the meme, predictions are coming out to be incorrect. On manually analyzing such examples we figured that it is important to take context from all parts of a meme's image to predict correctly. In the models like MMBT where output of Resnet's pooling layer is being used as image encoder, multiple parts of meme are not being treated separately. Although pooling is done by dividing the image into grids, these grids are fixed and not being learned. Essentially, multiple images in meme are being treated as a single image and context from multiple image is not getting captured for hate detection.

Difficulty in multi-image format
"*Different type of stations*"
True label: 0 Predicted label: 1

Inability to capture world knowledge
Though we pretrain the models on large amount of data, it still is difficult to incorporate every detail and nuanced relationships that exists in the world, in our models. Similar concern was observed in many examples while performing error analysis. The predicted result didn't match with ground truth result because it requires the model to know highly specific and contextual details. One of the examples for given scenario was observed on MMBT model results where the meme contains text : *"everyone is afraid of dark"*. As a human observer it is apparent that meme is referring to racial hate but our model fails to capture this world knowledge.

Inability to capture world knowledge
"*everyone is afraid of dark*"
True label: 1 Predicted label: 0

8.3 Errors on Hateful Memes, due to intermediate Finetuning on Sarcasm.

Intermediate fine-tuning on sarcasm was observed to drop our final fine-tuning accuracy on hate detection. Here, we try and identify what are the additional type of errors occuring due to this intermediate fine-tuning.
Sarcasm is increasing dependency on contrast: As seen in the error analysis for saracsm, it depends highly on cross-modal contrast to perform sarcasm detection task but, it doesn't always work when it comes to hateful memes. We can easily find several examples where the intermediate fine-tuned model failed to predict the correct output, potentially because it couldn't find the contrast between the text and the image tokens. For example, the image text: *"your order comes to $37.50 and your white privilege discount brings the total to $37.50"* with a white woman in the image. The intermediate trained model incorrectly labelled it as non-hateful.

Sarcasm is increasing dependency on contrast.
"*and your white priviledge discount brings the total to $37.50*"
True label: 1 Predicted label: 0

9 Contributions of group members

Overall, all members did roughly equal amounts of work and were always available when needed. The contributions of all our group members are listed below:

- Aishwarya: Baseline implementations (unimodals and multimodals with finetuning); error analysis; report.
- Shivam: Baseline implementations (unimodals and multimodals without finetuning); error analysis; report; proof reading.
- Dhruti: Learning MMF framework and performing various finetuning/ intermediate finetuning experiments on top of it; error analysis; report.
- Kriti: Learning MMF framework and performing various finetuning/ intermediate finetuning experiments on top of it; error analysis; report.

10 Conclusion

This project was a great first foray into exploring multimodal classification problems, which are highly relevant and hold practical importance in the present age of social media. We primarily tried our hand at the live NeurIPS competition of Hateful Meme detection, thereby appreciating what makes this problem truly challenging! Throughout our experiments and explorations, we learnt several things about this problem and about ML and NLP in general.

One of the crucial learning was to understand the drawback in the data collection and labeling technique used for sarcasm dataset. Using automated tools for data collection might lead to labeling errors. We explained examples for this in error analysis. Additionally, removal of unimodal biases before moving to multimodal implementations is required to accurately gauge multimodal model's performance. We also understood the importance of cross-modal dependencies in making predictions.

Intermediate fine-tuning helped us understand that choice of intermediate task is not as straightforward as looking for task similarity. It depends on multiple factors like the quality of dataset. We observed the drop in false positive rate after intermediate fine-tuning on saracasm dataset, validated by increase in AUC-ROC score.

The state-of-the-art multimodal models were observed to perform poorly on Hateful Meme dataset, with a large gap to human performance. As seen in the error analysis, this is potentially due to several models' shortcomings; such as the inability to capture world knowledge, failing to be robust over different image and text formats (multiple images and text split up), inability to unlearn biases etc.. This highlights the challenge's promise as a benchmark for multimodal research.

One thing that we wanted to try was experimenting with LXMERT and D&R architecture. Given more time we will try to make feature extraction work and runt he experiments on above architectures.

References

Castro, S., Hazarika, D., Pérez-Rosas, V., Zimmermann, R., Mihalcea, R., and Poria, S. (2019). Towards multimodal sarcasm detection (an _Obviously_ perfect paper). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4619–4629, Florence, Italy. Association for Computational Linguistics.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2018). BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Kiela, D., Bhooshan, S., Firooz, H., and Testuggine, D. (2019). Supervised multimodal bitransformers for classifying images and text. *CoRR*, abs/1909.02950.

Kiela, D., Firooz, H., Mohan, A., Goswami, V., Singh, A., Ringshia, P., and Testuggine, D. (2020). The hateful memes challenge: Detecting hate speech in multimodal memes.

Li, L. H., Yatskar, M., Yin, D., Hsieh, C.-J., and Chang, K.-W. (2019). Visualbert: A simple and performant baseline for vision and language.

Li, L. H., Yatskar, M., Yin, D., Hsieh, C.-J., and Chang, K.-W. (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.

Lu, J., Batra, D., Parikh, D., and Lee, S. (2019). Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks.

Pruksachatkun, Y., Phang, J., Liu, H., Htut, P. M., Zhang, X., Pang, R. Y., Vania, C., Kann, K., and Bowman, S. R. (2020). Intermediate-task transfer learning with pretrained models for natural language understanding: When and why does it work?

Ren, S., He, K., Girshick, R. B., and Sun, J. (2015). Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497.

Singh, A., Goswami, V., Natarajan, V., Jiang, Y., Chen, X., Shah, M., Rohrbach, M., Batra, D., and Parikh, D. (2020). Mmf: A multimodal framework for vision and language research. https://github.com/facebookresearch/mmf.

Yitao Cai, H. C. and Wan, X. (2019). Multimodal sarcasm detection in twitter with hierarchical fusion model. In *Proceedings of ACL*, page 2506–2515.