Hateful Memes
Meme Squad: CS 7643

Avijeet Ranawat* , Nagesh Kopparam*
Georgia Institute of Technology
{aranawat3, nkopparam3}@gatech.edu

Abstract

In this internet-driven era, content is the king, and its understanding, management, and classification have been the epitome of challenges faced by social media giants like Facebook, Twitter, Reddit, Instagram, etc. They all have this one thing in common —narrowly defined images overlaid with text, designed to spread from person to person via social networks, often for perceived humorous purposes — Welcome to the World of Memes.

With memes, the challenge today is harder than ever as both photos and text need to be understood together to get to know the real intent of the post. The original work by the Meta team had put out a baseline solution to classify the memes into hateful or non-hateful through a multimodal approach. In this report, we intend to give a walkthrough of our take on this task along with the challenges faced and resolutions made while implementing the models. Multiple models from scratch with late fusion and multiple multimodal models via the MMF framework were created for classifying memes. Our final MMF VisualBERT model was able to outsmart the benchmark score with an AUROC score of 0.8279 on test unseen data.

1. Introduction/Background/Motivation

1.1. Motivation

Today's world is filled with information from all over the world, we cannot begin to imagine how our lives would be without the internet. The world as we experience around us is not just a single sensory or unimodal but a combination of multiple senses (Vision, Sound, Touch, Taste, Smell). While smell, touch, and taste are still far-fetched on the internet, vision and sound are already present. And mostly it is a combination of visual and language (read/hear) on the online side.

As we use a combination of 2 abilities to interact online, naturally it would be evident to build new-age multimodal models which combine both computer vision and NLP tasks to make the machine understand things better. A few use cases of multimodal approaches are image captioning, and visual question answering (VQA). Many practitioners believe that multimodality holds the key to problems as varied as natural language understanding [7], computer vision evaluation [4], and embodied AI [11].

The plethora of data that we get exposed to daily can contain information that may be mean to a certain group of people. Our problem set here (hateful memes) is a good spot to test the capabilities of present-day classifiers to capture such hateful content and remove them before they are circulated among the masses.

According to Facebook's CTO, they acted upon 9.6 million pieces of content for violating their Hate Speech policies in the first quarter of 2020 [14]. This amount of malicious content cannot be tackled by having humans inspect every sample. Consequently, machine learning and in particular deep learning techniques along with continuous feedback loops (ex. report a post by the masses) are required to alleviate the extensiveness of online hate speech.

While initially the problem was tackled by fusing the word embedding from one model and vision embedding from another model and passing them to a fully connected layer at the end (late fusion) which provides the classification. This practice works only for examples that contain hateful text or hateful images or both. But when faced with examples that do not contain hateful content individually, but their combination turns out to be mean, our unimodal approach fails terribly.

Consider, as an illustration [Figure 1] taken from original paper [8], a sentence like "you smell nice today" paired with an image of a skunk, or "look how many people love you" with a picture of a tumbleweed in the desert. Unimodally, these examples are boring and harmless, but when the modalities are combined the meaning changes and they suddenly become mean/hateful—which is easy for humans to detect, but (so far) challenging to AI systems. It is dif-

* Random order. All authors made equal contribution

	Total	Not-hate	Hate	MM Hate	UM Hate	Img Conf	Txt Conf	Rand Benign
Train	8500	5481	3019	1100	1919	1530	1530	2421
Dev seen	500	253	247	200	47	100	100	170
Dev uneen	540	340	200	200	0	170	170	0
Test seen	1000	510	490	380	110	190	190	130
Test unseen	2000	1250	750	750	0	625	625	0

Table 1. Dataset splits

Figure 1. Multi-modal "mean" memes and benign confounders. Mean memes (left), benign image confounders (middle) and benign text confounders (right).

ficult and requires subtle reasoning yet is easy to evaluate as a binary classification. The task has obvious direct real-world applicability and cannot be solved by only looking at the image or the text, instead of requiring sophisticated multimodal fusion.

1.2. Dataset Description

The dataset was created by Meta for Hateful Memes Challenge present on the DataDriven site with the sole purpose of accelerating the research in the Multimodal domain. The dataset consists of exactly 10k memes. These memes can be labeled as either hateful or not hateful. The dataset is split into three sets: a train set of 8,500 samples, a dev set of 500 samples, and a test set of 1,000 samples. In addition to this "seen" test set, another test set consisting of 2,000 samples called the "unseen" test set, is available.

The dataset was created to encourage and measure truly multimodal understanding and reasoning of the models. A key point to achieve this are the so-called "benign confounders" (also called contrastive [3] or counterfactual [6] examples) which address the risk of exploiting unimodal priors by models: for every hateful meme, there are alternative images or text that flip the label to non-hateful.

For example, we can consider two scenarios with the same image but with 2 different text embeddings. Let the image be of an empty desert (Figure 1) with the 1st caption being "how many people love you" vs the second caption reading "how many people hate you". In this, the first meme turns out to be hateful, but the second one turns out to be non-hateful. Such image and text confounders require multimodal reasoning to classify the original meme and its confounders correctly.

The dataset construction procedure consisted of four phases: 1) data filtering; 2) meme reconstruction; 3) hatefulness ratings; 4) benign confounder construction. Using the various phases outlined and after further filtering to remove low-quality examples. Five distinct types of memes were formed: (a.) multimodal hate; where benign confounders were found for both modalities; (b.) unimodal hate where one or both modalities were already hateful on their own; (c.) benign image; (d.) benign text confounders and finally (e.) random not-hateful examples.

2. Approach

We were skeptical of the claims given by the original paper [8] and wanted to check from our own side if the unimodal approach fails so terribly as was suggested by the paper. For that analysis, which we call the base model in our case, we fetched the Hateful Memes dataset of 10k memes that had originally contained varied-size images. Resized the images to a common 224 size so that it would be easy for the convolution models to create feature maps from them. Then transformed them into RGB tensors. To create our image embeddings, we utilized torchvision, by passing 2048 as an input dimension.

The text part of the image was already available to us in the dataset with each image id having the corresponding text associated with it. So, there was no need to capture text from images with any fancy OCR approach. We fetched this text part and passed it to our fasttext module for generating the embedding from it. The fasttext internally utilized the CBOW technique for generating these tensors.

Further, we combined these two components of our data, while also making sure that the dataset is balanced in the sense that both hateful and non-hateful content are present in almost equal proportions. For the base model, we combined these embeddings which we got separately from fasttext and torchvision modules to get a fusion embedding output of dimension 512. And this was intern passed as an input to the fully connected layer of the pre-trained RESNET152 model.

Adam optimizer and ReduceLROnPlateau optimizer from the pytorch-lightning part was used for the optimization of weights. Then the generated logits from the fully connected layer were passed to softmax CE for predicting the final binary classification of input meme data as output label of 1 being hateful or 0 being not hateful. Finally, the AUC ROC score was calculated for the test data available to us. Giving us the criteria for measuring the performance of different models.

To compare the effects of late fusion to early fusion, we used the MMF models which are part of a framework for vision-and-language multimodal research from Facebook AI Research (FAIR). MMF provides us pretrained out of the box SOTA models, and training dataset for multimodal analysis. Under the hood it is powered by Pytorch and helps us in modularized implementation of multimodal models. For every image, we extract certain boxes of 2048D region-based image features from a fully connected layer of a ResNeXT-152 based Mask-RCNN model [5]. We project the visual embeddings into the textual embedding space before passing them through the transformer layers. We learn weights to project the image embeddings to 768-dimensional token input embedding space. Specifically, the VisualBERT model was pre-trained on the CC dataset and combined various models to create an ensemble of models. They classified the test set based on the probability scores of the label being a hateful meme (1 for hateful, 0 for other).

3. Experiments and Results

3.1. Loss function and Training Scheme

The area under the receiver operating characteristic curve (ROC AUC) [1] has been selected as the measure of performance. The reason for selecting this metric was simple because we were dealing with a binary classification problem (hateful or not hateful labels), and this is the go-to metric for it. AUROC score is given by the following formula:

$$AUROC = \int_{x=0}^{1} \mathrm{TPR}\left(\mathrm{FPR}^{-1}(x)\right) dx \qquad (1)$$

The aim was to increase the AUROC score and reduce the validation loss incurred while training. The AUROC score gives us a fine-grained sense of classifier performance, although the accuracy metric will also work in our case since it is easily interpretable and the dev test sets are not extremely unbalanced (Table1). The choice of ROC curve would be better as it gives us a much more detailed view on the classifier cutoffs compared to having one cutoff in case of an accuracy metric.

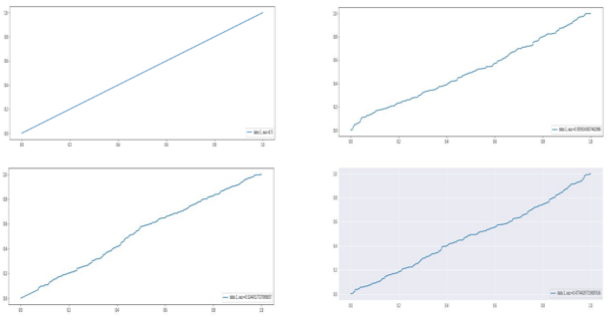

Figure 2. AUCROC curves along with the scores. (a.) Top left 0.50, (b.) Top right 0.505, (c.) Bottom left 0.524, (d.) Bottom right 0.509

3.2. Experiments and Results for Base model

In this context, we call the implementation of late fusion of visual language unimodal models from scratch without using the MMF framework as a base model. Our aim with implementing the late fusion models was to evaluate how far off the performance is when compared to the SOTA MMF frameworks. Additionally, the late fusion models did not use any of the SOTA models like Image Grid, Image-Region, and Text BERT which were the core for unimodal analysis in the MMF framework and the original paper [8].

In terms of architecture, our first try had 150 as its hidden embedding dimensions, and 300 as its language features dimensions, with the same amount of vision feature dimensions. The reason to choose these values was that the more dimensions we have in the embedding layer the better the model would be. But then there must be a limit to it for faster training, so selected these values. The combined fused output of size 256 and some comparative large learning rate of 0.05, since it was more of a first try, were also tested for only 5 epochs. It was destined to fail and gave us the perfect straight line in the AUCROC curve [Figure 2 a.] and an AUROC score of 0.5 (a random model). That was our first try and it predicted every test image as a hateful meme. The next steps were to increase the epochs and decrease the learning rate to 0.00005. These changes helped provide more time for the model to train the hidden features and as expected it delivered some increase in the score [Figure 2 b].

Moving on towards the next try we experimented with image size, dropout, etc. The image dimension was increased by 500 and the dropout rate to 0.5 (from the earlier 0.1). We increased the dropout to strengthen the regularization effect on our model to avoid over-fitting. To speed up things and parallelize training, we increased the number of workers to 16 from an earlier of 4. These changes did not help in improving the results (AUROC score - 0.5059143). A marginal improvement, but miles away from the SOTA models.

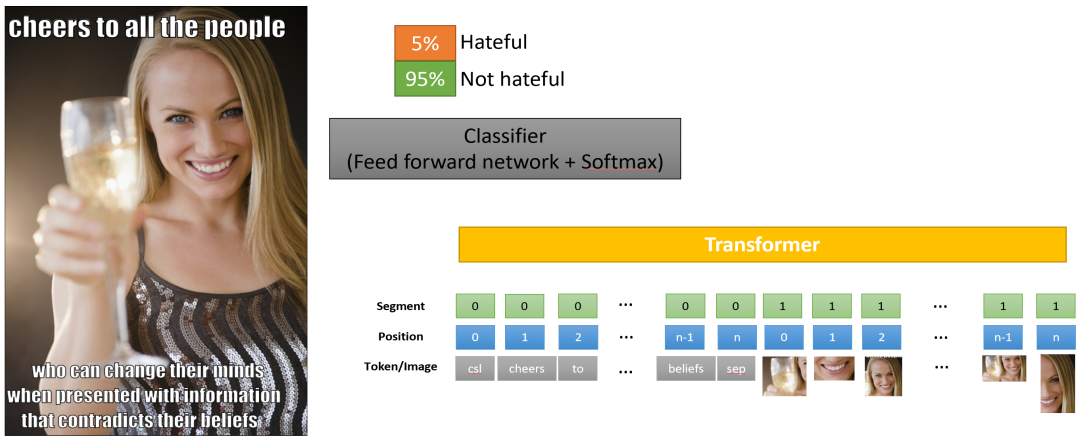

Figure 3. An example meme sampled from the dataset (left), and an illustration of the multi-modal transformer architecture (right). Image regions and language are combined with a Transformer to allow the self-attention to discover implicit alignments between language and vision.

In our next step we were curious about the image dimensions and experimented on it. How would the changes to the image dimensions help/harm our model? Turned out that reducing the dimensions by a considerable extent we improved the AUROC results to around 0.524431 [Figure 2 c.] and increased the speed of our execution as well. That was some good improvement with one change. For the next iteration, the batch size was increased to 256 but it was an overkill as our AUROC scores dropped again towards 0.509115 [Figure 2 d.].

In the new wave of changes, the epoch size was increased to 100 to help the model provide a better chance to learn the nuances in the data. An early stopping along with patience feature was also used to stop the training in case there is no decrease in the loss/ adverse loss effects. Turns out that early stopping worked and halted the training early. Even after trying multiple other iterations, the best AUROC score achieved by late fusion models was 0.53. These series of experiments clearly solidify the statement by the original authors that the Hateful Memes dataset is not solvable by late fusion unimodal models, but to fine-tune, and test large-scale, pre-trained multimodal models. That may be accounted for due to the size of the dataset (10K images) which is insignificant compared to datasets such as Visual Genome (108K) [9], COCO (330K) [2], and Conceptual Captions (3.3M) [13]. These series of experiments and learnings gave our base model a stop and we headed straight towards mighty SOTA MMF multimodal analysis.

3.3. Experiments and Results for VisualBERT

Our next milestone was to experiment with VisualBERT [10] – which meant to be the "BERT of vision and language" – that was trained multimodally on images and captions. The main reason to choose VisualBERT was because of its benchmark [Table 3] results given by [8]. Now given the benchmark result of 71.33 AUROC on test for VisualBERT and a score of 71.41 AUROC on test with VisualBERT COCO pretrained on COCO dataset. These benchmark results are already much better than our base model implementation without MMF. We wanted to outperform such a good result with our approach and were up for a tough challenge. Certainly, working with the original data would not get us far enough. That is why we considered increasing the dataset size for training by including 428 additional memes from the Memotion Dataset [12] containing 14K annotated memes with human-annotated labels, with inspiration from [15] for our approach.

This approach can be divided into four sections: dataset expansion, image encoding, training, and ensemble learning. Figure 3 illustrates an overview of the architecture.

VisualBERT is originally pre-trained on COCO image caption dataset, but in our experiments, we noticed that the model pre-trained on Conceptual Captions (CC) achieves noticeably better scores. Therefore, we picked the latter model which is provided by MMF. We fine-tuned the pre-trained VisualBERT model on the aggregated training set and evaluated it on dev unseen and test unseen datasets.

We use the first output of the final layer as the input to a classification layer clf $(x) = Wx + b$ where $W \in \mathbb{R}^{DxC}$, with D as the transformer dimensionality and C as the number of classes(see Figure 3). We apply a softmax on the logits and train with binary cross-entropy loss.

For the first iteration we choose the 3000 as our max_update (epochs), with learning rate of 5.0e-05 and a batch size of 32 for the parameters, resulting in an AUROC of around 0.8112. For the second iteration we increased the batch size to around 80, along with max_updates to 3500 keeping the learning rate fixed, that resulted in a score of

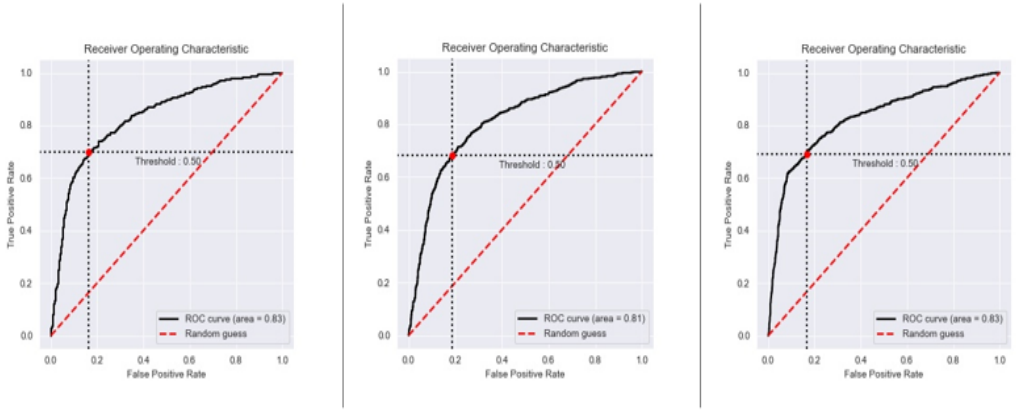

Figure 4. AUCROC curves for VisualBERT MMF model along with scores on test data: AUROC score 0.8214 (left), AUROC score 0.8161 (middle), AUROC score 0.8279 (right) for different iterations tried. In the curves middle line depicts the random model performance, while the black line shows the trained model predictions, and red dot depicts the threshold selected.

		Validation		Test	
Type	Model	Acc.	AUROC	Acc.	AUROC
Baselines	VisualBERT	62.10	70.60	63.20	71.33
	VisualBERT COCO	65.06	73.97	64.73	71.41
Ours	Best model	71.03	77.16	78.42	82.79

Table 2. Model performance

around 0.8261. Further we experimented with different values for max_update, batch size, learning rate, input output dimensions of embeddings and hidden layers. Eventually we found the best model to be giving the score of 0.8279 AUROC for test data and was generated by applying ensemble learning, which created one strong model from multiple 'weak' models to get the final AUROC score. There is still lots of room for improvement in this project and further research teams can focus on ways to improve the fusion of modalities, so that the models are able to understand the true connection between the text and the image part of memes.

4. Challenges faced/ Experience

The challenges started from the very beginning of our exploration. We were mixing and matching the two datasets, in that we had to do manual checks for whether the Memotion dataset was correctly labelled as per in alignment with hateful memes dataset. MMF also required the dataset in a specified format, so unzipping the data to that given format also took us time. Also, our initial base model failed terribly, though it was mentioned in the challenge that it has benign confounders which may hamper the results, but we took the challenge head on to learn the reality. Moreover, it was our first deployment on GCP and we faced a lot of issues related to VM, permissions, transferring data in out of the VM and these huge datasets along with enormous sized pretrained weights were tough to deal with. We also ran into issues related to low memory and disk space when experimenting with higher batch sizes the lower configuration of 15GB RAM. Though we were able to increase our VM configurations, it took us a bit of learning.

In terms of coding, we faced multiple version conflict issues for old and new pytorch lightning libraries and its compatible versions of torch-text and torch-visions. There were multiple instances where after running the code for an hour the colab got stuck and had to restart the training from scratch, wasting the limited time and resources we had. But then these challenges are part of every project and lots of learnings hidden in them.

5. Work Division

Both of us made equal contributions and made the best use of the limited resources that we had with time and GCP credits. The details can be seen in Table 3.

Student	Contribution Aspects	Details
Avijeet	Paper review Experimentation Analysis Documentation	Paper review GCP setup and test Original dataset preparation Base multimodal model with late fusion Experimentation on base model AUCROC curves Reports Live coding & peer review
Nagesh	Paper review Experimentation Analysis Documentation	Paper review GCP setup and test Additional dataset preparation MMF multimodal models Experimentation on MMF Architecture diagram Reports Live coding & peer review

Table 3. Team Contribution

References

[1] Andrew P Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition,30(7):1145–1159*, 1997. 3

[2] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 4

[3] Matt Gardner, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, and et al. Evaluating nlp models via contrast sets. *arXiv preprint arXiv:2004.02709*, 2020. 2

[4] Donald Geman, Stuart Geman, Neil Hallonquist, and Laurent Younes. Visual turing test for computer vision systems. *Proceedings of the National Academy of Sciences, 112(12):3618–3623*, 2015. 1

[5] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. *In Proceedings of the IEEE international conference on computer vision, pages 2961–2969*, 2017. 3

[6] Divyansh Kaushik, Eduard Hovy, and Zachary C Lipton. Learning the difference that makes a difference with counterfactually-augmented data. *arXiv preprint arXiv:1909.12434*, 2019. 2

[7] Douwe Kiela, Luana Bulat, Anita L Vero, and Stephen Clark. Virtual embodiment: A scalable long-term strategy for artificial intelligence research. *arXiv preprint arXiv:1610.07432*, 2016. 1

[8] 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:2005.04790*, 2020. 1, 2, 3, 4

[9] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, and et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision, 123(1):32–73*, 2017. 4

[10] Liunian 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. 4

[11] M. H. Ribeiro, P. H. Calais, Y. A. Santos, V. A. Almeida, and W. Meira Jr. Characterizing and detecting hateful users on twitter. *In Twelfth International AAAI Conference on Web and Social Media*, 2018. 1

[12] Chhavi Sharma, Deepesh Bhageria, William Scott, Srinivas PYKL, Amitava Das, Tanmoy Chakraborty, Viswanath Pulabaigari, and Bjorn Gamback. Semeval-2020 task 8: Memotion analysis–the visuo-lingual metaphor! *arXiv preprint arXiv:2008.03781*, 2020. 4

[13] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. *In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2556–2565*, 2018. 4

[14] Tekla S.Perry. Qa: Facebook's cto is at war with bad content, and ai as his best weapon., 2020. 1

[15] Riza Velioglu and Jewgeni Rose. Detecting hate speech using multimodal deep learning. *arXiv preprint arXiv:2012.12975*, 2020. 4