Identifying Hateful Memes with Multimodal Classifications

Alex Fan
Department of Statistics,
Stanford University
alexfan@stanford.edu

Yixuan (Sherry) Wu
Department of Statistics
Stanford University
sherryyw@stanford.edu

Abstract

Hateful meme detection is not only fundamental in governing a healthy online environment, but it is also an important research topic that requires both visual and linguistic modeling skills and advanced knowledge of effectively combining multimodal representations. In this paper, we utilized data from Meta's Hateful Meme Detection Challenge, which contains examples of memes that can be successfully classified only when their text and images are considered coherently.

We built three models, a baseline, a VisualBERT, and a VisualBERT with external feature extraction (Model 3). By leveraging large pre-trained models and fine-tuning, our best model, Model 3, achieves a 62.4% accuracy. We presented our findings, analyzed and compared the results both quantitatively and qualitatively, and discussed potential future steps.

1. Introduction

Since the emergence of MySpace, social media has experienced tremendous growth and has reshaped our ways for communications. [19] Any user can share, comment, and react to any other users, unlimited by physical distance. While individuals' opinions are more easily visible to others and can thus provide inspirations and entertainment, social media has also inevitably provided an expressway for hateful expressions to propagate, which in turn fuels provocations of anger, hatred, and extremism.

The effort to detect and thus strike down hate speech on social media is not new. Hateful conduct policies of social media platforms have constantly been maintained and updated. Besides, around 18,800 scholarly articles published with keywords of "hate speech detection" and "social media" since 2005. [1] More recently, there has been great progress in using deep learning-based models, and more specifically transformer-based models, to detect hate speech. For example, AI-BERT+CNN received 0.9, 0.79, and 0.97 F1 scores on the Davidson, Founta, and Twitter Sentiment Analysis data respectively. [16]

Despite the promising progress, we need to acknowledge that other forms of hateful conduct are also prevalent on social media and require different modeling. While most forms are unimodal, such as hate speech, video, and audio, a more unique form is memes, which use both image and texts. According to Iloh [8], memes are integrated in daily communication, serving as a symbolic reflection on cultures and communities. Moreover, they are an integral part in younger generations' communications, as 55 percent of 13-35-year-olds share memes every week [2]. Therefore, it is important to deter hate spreading through memes.

Because memes contain both images and texts, it has been incredibly hard to accurately detect and classify hate memes. The main challenge present for training a hate-detection model is for it to understand the coherence between the corresponding text and image in any given memes. As demonstrated in Fig. 1 from Kiela et al. [12], the images and texts of memes often may appear innocuous independently, but when taken together they have hateful implications. Furthermore, unlike typical visual-linguistic objects, memes can contain subtle visual cues that ultimately determines their polarity. In addition, memes are likely to contain references to specific and diverse contexts that are not commonly used, making learning more challenging [27]. As a result, multimodal approach that effectively bridges CV and NLP is necessary.

Our project aims to build and train a model that, given an input - a meme (which contains both words and an image) can output a binary classifier that classifies the meme as either hateful or not. We utilize the data from Facebook's Hateful Meme Challenge [12]. Our baseline model is multimodal with independently pre-trained text embeddings (sentence-BERT, [21]) and visual embeddings (ResNet152). We also build a VisualBERT standalone model and improved on it by adding external feature tags.

2. Related Work

Deep learning methods used to detect hate in memes can be roughly divided into three categories: unimodal models,

Figure 1. Example of memes with their hateful/non-hateful properties determined by the visual and text coherently.

multimodal models with unimodal pre-trainings, and multimodal models with mutlimodal pre-trains.

Under the unimodal category, the most notable are ResNet [7], Faster R-CNN [22], and BERT [5], with the former two as image classifiers and BERT as a transformer for language. The architecture of both ResNET and Faster R-CNN have been covered in class, and a detailed description of BERT is in the Method section below. In general, advantages of unimodal models is that there have been well-established research on these methods and implementations are thus straightforward. However, they are inadequate in our task of correctly classifying memes which contain two modalities. Therefore, given the time constraint, we decided to not implement any model from this category.

Multimodal methods with pre-trained image and text contain various models. Ones with simpler structures include models using simple fusion, where pre-trained text and image embeddings are concatenated. Other model have more complicated structures. For example, supervised multimodal bitransformers [11], which uses an image encoder that accept arbitrary number of inputs and outputs separate image embeddings. These image embeddings are then projected onto text token space with a learnable weight matrix. Models in this category include ViLBERT [15], Vl-BERT [24], and VisualBERT [13], which were all submitted around August 2019. Compared to the unimodal category, these models analyze both text and image representation of a meme, and thus are more capable of determining the coherence between text and image. Moreover, these models are independent of feature extractions and thus can be applicable to different tasks [11].

Other methods use multimodal pre-training. Notably, the difference between this and the previous categories is mainly in the pre-training. Therefore, the backbone models remain largely unchanged. Models in this categories include Visual BERT trained on the COCO (Microsoft Common Objects in COntext) dataset that contains over 330K images suitable for image captioning [14], and ViLBERT with Conceptual Captions pre-training, which contains images and image caption styles that are more diverse than the original COCO set [23]. An advantage of these models is that the information of text and image is already integrated through the pre-training, and thus their relationship can be more accurately captured. However, while multimodal methods with unimodal pre-trainings can edit text or images encoders independently without having to retrain multimodally, the process for those with mutlimodal pre-trainings is less efficient. Despite such shortcoming, models in this category have been shown to be more effective in classification [12].

While these three categories capture the three main approaches to our problem, other models on the Leaderboard for the Hateful Meme Challenge explored additional options. Zhu topped the leaderboard with an unseen AUROC of 84.50 and accuracy of 73.20 by using external labels such as race tags for face and head [27]. Muennighoff was the 2nd place by ensembling five different multimodal models, achieving 83.10 and 69.50 AUROC and accuracy, respectively [17]. While these methods are effective, our goal by completing this project is to gain learning experience of building multimodal models, and therefore, we focused on more understanding and re-implementing the multimodal models described in the previous two paragraphs.

3. Method

3.1. Multimodal Classification Overview

Multimodal classification has become an increasing important and popular area of research in the last decade. Combining data representations from different modalities, it provides a much more comprehensive analysis and understanding of current problems spanning many domains [9].

3.1.1 Data Fusion

A critical piece in working with multimodal data is how to combine drastically different representations into a coherent joint representation. For example, in Fig. 2, while the visual representations is dense and high-dimensional, the text representation is sparse. Therefore, an effective fusion algorithm is necessary to conquer this challenge. Fusion algorithm has evolved drastically during the past few years. Our paper will focus mainly the conventional fusion steps, which were used in both our baseline and VisualBERT models.

Conventionally, the fusion step can occur in different stages of the classification process. Fusion methods can be categorized under either the model-agnostic approach, which are independent of the specific models, or the model-based approach [3]. In the model-agnostic approach, there are three commonly used approaches: early, late, and cross-modality fusion (see Fig. 3)

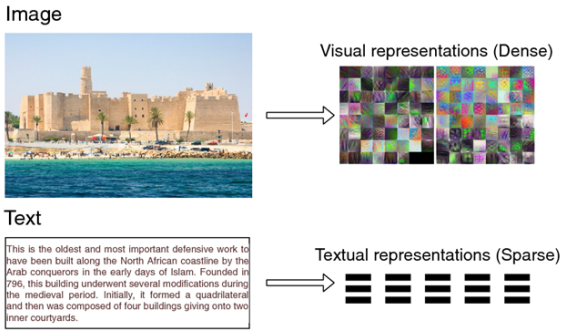

Figure 2. Challenge in combining different representations [18]

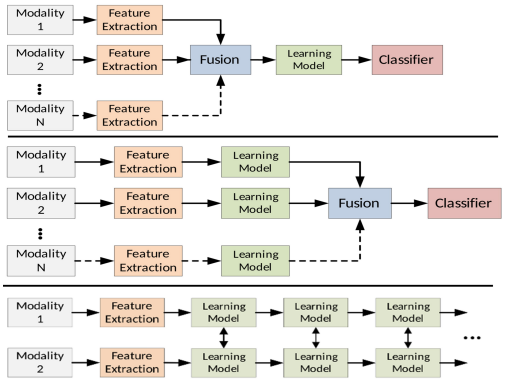

Figure 3. From top to bottom: early fusion, late fusion, cross-modality fusion

In early fusion, features of data are independently extracted from different modalities. This fusion occurs on the feature level, and thus learning occurs on the joined data. A disadvantage of this method is that fusing different modalities into one single representation can result in lack of homogeneity and thus larger prediction error [18].

In late fusion, each data source undergoes feature extraction and learning independently, followed by fusion at the decision-making stage. However, because of such independence, this method may fail to integrate data from different modalities well. [18]

In Cross-modality fusion, however, fusion can occur during different stages of model training. Specifically, each modality can utilizes information from other modalities to improve performance. However, outputs from different streams may result in different shapes, thus increasing the difficulty of fusing.

3.1.2 Multimodal representation

Another challenge in multimodal classification which is also relevant for our project is how to represent the data for computational purposes. Baltrusaitis et al. [3] introduced two categories of representations, joint and coordinated representations (Fig. 4). We will focus mainly on joint representation.

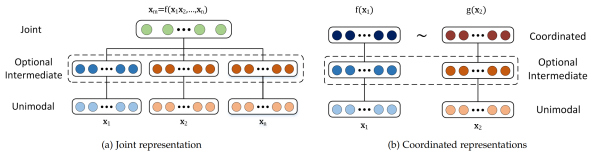

Figure 4. From top to bottom: early fusion, late fusion, cross-modality fusion

Concatenation is a basic joint representation method. Our baseline model uses feature concatenation, which puts features into one single vector (detailed below). Another category of methods use neural networks. For example, building on feature concatenation, Deep Concatenation combines features into a single deep learning layer. A more complicated method is Deep Merge, which is more commonly found in models with encoders. Each output node can take multiple input nodes across modalities. Lastly, Score Concatenation uses probability scores and such scores are combined with concatenation [9].

3.2. Baseline Model

The baseline model [1] used two pre-trained models for text and vision that were fused and fine-tuned to our specific task of classifying hateful memes. The image data was converted to a format suitable for torchvision models and fed through the pre-trained Resnet152 model, outputting an image embedding in $\mathbb{R}^{B*1000}$. For the text data, we leveraged S-BERT's [21] sentence embedding, which pools each word-level embedding of the sentence into a single sentence-level embedding. This outputs as a matrix in $\mathbb{R}^{B*768}$. The fusion method is a simple concatenation of the embedding dimension. Finally, the concatenated $\mathbb{R}^{B*1768}$ matrix is fed through a fully-connected linear and ReLU activation layer before projecting down to $\mathbb{R}^{B*1}$ for the softmax scoring. We use a binary cross-entropy loss in order to backpropogate the errors.

3.3. VisualBERT Model

3.3.1 Overview

A model extremely relevant to our project is VisualBERT, which is a joint representation model for vision and language [13]. VisualBERT is based on BERT [5], a Transformer that "pre-train[s] deep bidirectional representations from unlabeled text." VisualBERT reuses the self-attention mechanism from BERT, and building on BERT, it adds a set of visual embeddings for image modelling.

[1] All models were built using the Pytorch framework [20] and our starter code was adapted from Stanford cs224n's default project

The visual embedding is a sum of three embeddings: the first embedding is a visual feature representation constructed using CNN; the second embedding serves as an indicator to differentiate a given visual embedding from a text embedding; the third embedding takes the aligned words and images' bounding regions as input and outputs the sum of positional embeddings corresponding to the words aligned with the image.

The visual embedding is then concatenated with the text embedding, as shown in Fig. 5 and fed into the transformer. This allows model to learn about the alignment between text and images and thus build a reasonable joint representation. The rest of the structure follows that of the BERT.

While traditional BERT involves only one pre-training process, VisualBERT splits that into task-agnostic and task-specific pre-training. In our case, the task-agnostic pre-training is on COCO (introduced in Related Work section). Two procedures are done here: 1. the model is trained to predict some elements of masked inputs with vectors representing un-masked regions. 2. the model is trained to distinguish two captioning methods when more than one captions present for a given image; the first method contains two captions that corresponds to the image, while the second methods contains one corresponding caption but another randomly drawn caption. The task-specific training uses the target data with masked language modeling to familiarize the model with the specific task.

Followed by pre-training is fine-tuning, which closely resembles that of BERT. Provided with task-specific inputs and outputs, all the parameters are fine-tuned [5].

3.3.2 VisualBERT w/o Feature Extraction

The VisualBERT-based model for the Hateful Memes dataset relies on pre-trained VisualBERT weights, accessed using the Huggingface library [25]. The inputs to the VisualBERT pre-training step requires both visual embeddings and tokenized text. The text is fed through the BERT uncased tokenizer model, which is able to take into account out-of-vocabulary words to a certain extent by tokenizing based on word pieces. To make sure text inputs to the VisualBERT model are the same size, the BERT tokenizer pads the input up to a max length of 100 tokens with corresponding attention masks.

The visual embedding inputs for VisualBERT is fairly particular. Since the original model worked by aligning input text with regions within the image, its visual inputs are actually embeddings from an R-CNN [13]. Because of this, the raw pixels are initially fed through a detectron2 R-CNN model [26]. The particular configuration of the detectron2 model uses a Resnet101 model as the backbone and boxes were chosen based on an non-maximum suppression score criterion. The R-CNN [2] outputs a $\mathbb{R}^{B*100*1024}$ matrix which will be the visual embeddings for VisualBERT.

The visual and text inputs are used for the VisualBERT pre-trainer, which is based off the COCO dataset for a natural language and visual reasoning task. The last hidden state of this pre-trained model, which is in $\mathbb{R}^{B*200*768}$ is flattened and then fed through fully-connected feed forward layers with LeakyReLU activations. Layernorm and dropout regularization are also applied before and after the activations respectively. Finally, the matrices are projected down to $\mathbb{R}^{B*1}$ for softmax scoring in a binary cross-entropy loss function.

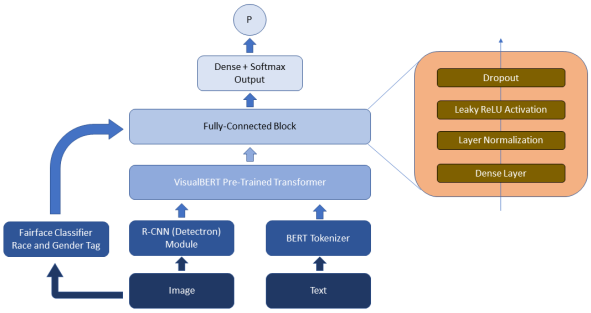

Figure 5. VisualBERT with FairFace Features Model Architecture

3.3.3 VisualBERT w/ FairFace Classifier

This model is a variation of the vanilla VisualBERT model, with added feature extraction using FairFace Classifier [10]. This version follows the same architecture of the vanilla VisualBERT model, but after obtaining the last hidden state from the VisualBERT pre-trainer, the race and gender features that were extracted are fused to the embeddings via concatenation. The fully-connected layers are adjusted accordingly to accommodate the extra features in the embedding matrix. This usage of FairFace is slightly different from the implemention of Zhu (2020) [27], but ideally it does not lose too much performance.

4. Data

The Hateful Memes dataset was pre-processed by Facebook AI, as detailed in Kiela et al. [12]. After reconstructing the memes, the team hired annotators to filter the memes that were duplicates, not in English, violating, or containing slurs (which are unimodal). Then for each meme, five annotators rated it as "definitely hateful", "not sure", or "definitely not hateful". The memes with disagreements were removed and the resulting label for each meme is binary.

[2]Our implementation adapts from the tutorial given by huggingface https://huggingface.co/docs/transformers/model_doc/visual_bert

To increase the challenge, the team added benign confounders, which are created by minimally replacing the image or text in a meme so the originally hateful meme becomes non-hateful (see Fig. 1).

Therefore, within the hateful category, memes can be unimodal hate, where the text and/or images could already be hateful, or multimodal hate, where only the combination of the text and the visual makes them hateful. Besides the non-hateful examples, the non-hateful category also contains benign confounders that, through either text or image replacement, become non-hateful.

The final dataset we used contains 11,040 memes, with 8,500, 540, and 2000 memes used as train, dev, and test set respectively. All sets are claimed to be balanced with the following compositions: 10% unimodal hate, and 40% multimodal hate; 20% benign text confounder, 20% benign image confounder, and 10% non-hateful. That being said, all splits were found to be unbalanced with usually more negative samples (e.g., the train split contain 5450 non-hateful vs. 3050 hateful). Therefore, at training time, the train set is manually balanced by sampling the negative samples such that their count equals the number of positive samples.

More specifically, in the train split, the text has a mean and median of 12 and 10 words respectively. The images have an average height of 597.65, a minimum and maximum of height are 94 and 825, an average width of 523.60, a minimum and maximum of widths are 95 and 823. Most of the models that were tested relied only on the raw pixel and text data. Some examples are shown in Fig. 1

The image data was preprocessed by resizing to a height and width of 224 and normalizing across all images in the split. Examples are shown in Sec. 4. This pre-processing is necessary for both the baseline and VisualBERT-based models. For VisualBERT, the detectron R-CNN backbone takes in BGR pixel data instead of the usual RGB, so that was converted specifically for that model.

There is some implicit feature extraction within the tokenization process and R-CNN backbone for the VisualBERT model. The only explicit feature extraction that was performed was in our VisualBERT with FairFace classifier model. We extracted gender and race tags from the images using FairFace classifier [10]. This extraction method first predicts a bounding box on any face in the image. Then it scores and predicts the race and gender (among other features) on each face bounding box. For each image, a binary race and binary gender flag was created. For race, if a non-white face was detected within the image, it was coded as 1, else for white or no race information it was coded as 0. For gender, if a non-male face was detected, the image was coded as 1, else it was coded as zero. These flags were fused with the embeddings to help the model understand the presence of minority populations.

5. Results

5.1. Experimental Details

Hyperparameters were chosen by validating on the development set provided. For the baseline model, the learning rate was chosen to be 1e-3 with a hidden size of 1200. A dropout regularization with probability 0.1 was included as well. For the VisualBERT models, the learning rate was 1e-5 and the hidden size was reduced to 1000. The VisualBERT models also used a dropout probability of 0.2.

For both models, training batch size was maximized to the available GPU memory, which resulted in a batch size of of 32. We chose to use the Adam optimizer due to its robustness; its parameters, β_1 and β_2 were 0.9 and 0.999 respectively. The models were run on AWS g4dn.xlarge instances, which are powered on single NVIDIA T4 GPU.

Performance was evaluated by computing accuracy and the area under the receiver operating characteristic curve (AUROC), which were calculated using sk-learn's API [4]. The AUROC is standard in terms of model comparison because it measures the because of its scale and classification-threshold invariance. The accuracy, on the other hand, gives a more interpretable quantity of correct hits relative to the sample size.

5.2. Quantitative Results

The results from these models show just how hard this problem of toxic meme classification really is. The VisualBERT-based model shows no improvements over the baseline simple concatentation fusion model despite being multi-modally pre-trained. The VisualBERT model with additional features from FairFace nominally had some improvement in accuracy (59.8% to 62.4%), however that improvement is largely because from being too conservative in labeling hateful memes, as is discussed in the qualitative analysis section.

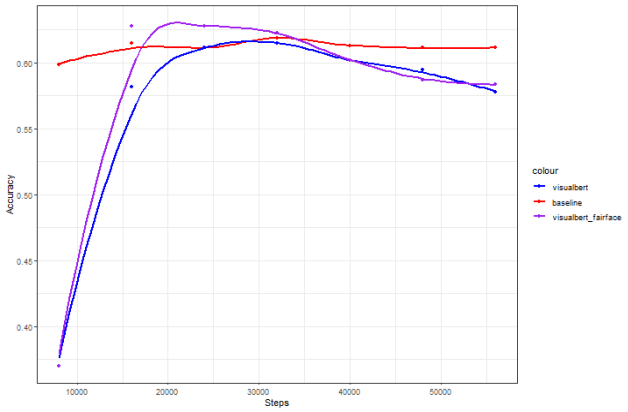

Figure 6. Model Dev Accuracies

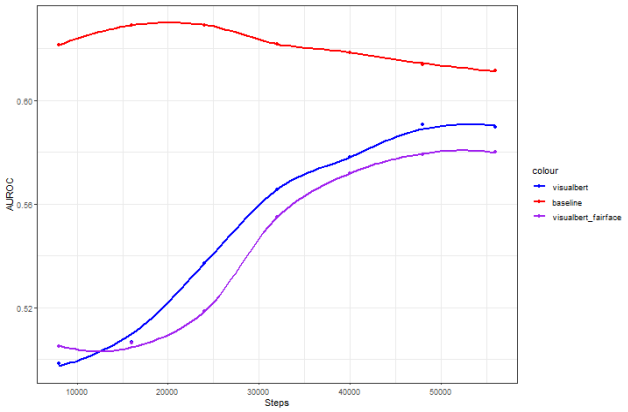

Figure 7. Model Dev AUROC Scores

Model	Acc.	AUC
Baseline	59.8%	60.5%
VisualBERT	59.1%	57.1%
VisualBERT w/ Feature Extraction	62.4%	57.4%

Table 1. Test Accuracy and AUROC for each model.

The models underfit the relationship between the input text/image data and the hatefulness of the meme, with the accuracy being barely better than randomly guessing. At the same time, increasing the number of training steps does not help the model converge either. This could be due to the relatively small number of training samples (6000). With the small sample size, the model is not able to fully learn the complex nature of memes such as the use of sarcasm. Alternatively, this underfit could also be explained by the model architecture and a feature space that is not rich enough.

5.3. Qualitative Results & Discussions

In this subsection, we will be analyzing mis-classified memes. Because of the nature of these memes, we want to make the reader aware of the hate-filled rhetoric before proceeding.

Classification performance is analyzed [3] using the confusion matrix, as shown in Tab. 2, Tab. 3, and Tab. 4.

Trends that are consistent across three models are: (1). Each model classifies a given meme as Non-Hateful much more often (64.7%, 71.4%, and 85.35%, respectively). (2). With the hypothesis that a given meme is hateful, each model is more prone to have Type-I error (wrongly classifying a meme as non-hateful, bottom-left corner of a table) than Type-II error.

Trends that are specific to the model: (1) While VisualBERT w/o feature extraction (**Model 2**) has a better overall accuracy compared to the baseline model (**Model 1**), it correctly classifies hateful memes less successfully. In fact, VisualBERT with feature extraction (**Model 3**) is even more prone to this issue compared to Model 2. (2) Similarly, Model 3 is more prone to Type-I error compared to Model 2, which is more prone to Type-I error compared to Model 1. The decreasing trend of AUROC is likely because false positive and negative rates become more unbalanced from Model 1 to 3.

Predicted Actual	Non-Hateful	Hateful
Non-Hateful	870 (43.5%)	380 (19.0%)
Hateful	424 (21.2%)	326 (16.3%)

Table 2. Confusion matrix of Baseline model

Predicted Actual	Non-Hateful	Hateful
Non-Hateful	944 (47.2%)	306 (15.3%)
Hateful	484 (24.2%)	266 (13.3%)

Table 3. Confusion matrix of VisualBERT w/o feature extraction

Predicted Actual	Non-Hateful	Hateful
Non-Hateful	1202 (60.1%)	48 (2.4%)
Hateful	705 (35.25%)	45 (2.25%)

Table 4. Confusion matrix of VisualBERT w/ Feature Extraction

To gain a better understanding of the trends discussed, we look into and divide misclassified images into some subcategories and provide examples in these sub-categories. Since we have 4^3 potential categories (each model has 4 categories) and even more when combining them, we decided to only look at certain important and major categories.

[3] Startup code including data read-in is provided here: [6]

(1). Images that only Model 3 classified correctly (see Fig. 8, 221 such cases). It is possible that some of these memes contain face with the correct racial tags. Specifically, the bottom two demonstrated how otherwise neutral texts paired with images with certain races are detected by Model 3 as hateful.

Figure 8. Examples that only Model 3 is correct. The top 2 are actually non-hateful, while the bottom 2 are actually hateful

(2). Images that Model 3 classified wrong but Model 2 classified correctly (see Fig. 9, 260 such cases). It is thus important to note the caveat of adding face tags here. For example, the top left may be wrongly classified as hateful because of its racial feature. Similarly, the racial tag is likely to be inaccurate with the lighting and number of people.

Figure 9. Examples that only Model 2 is correct, but Model 3 is wrong. The top 2 are actually non-hateful, while the bottom 2 are actually hateful

(3). Images that all models wrongly classified as non-hateful (see Fig. 10, 350 such cases). There are It seems that the models cannot recognize offensive images while the texts look neutral. One potential reason is that we don't have enough images during training for the model to learn that certain person's face (Hitler's, for example) often lead to hateful contents. Granted, it is incredible difficult for machine to learn about certain reference (Planet of Apes reference here and its relationship with Floyd, for example).

Figure 10. Examples of Wrong Non-Hateful classifications

(4). Images that all models wrongly classified as hateful (see Fig. 11, 15 such cases). It seems that models deem many memes with offensive language ("killing" for example) and various profanity as hateful despite the meaning the memes are conveying are neutral or positive.

Figure 11. Examples of Wrong Hateful classifications

In conclusion, misclassifications above demonstrate that the models have difficulties understanding some images alone, and still have difficulties drawing meaningful connection between a given image and text. A potential solution for the former issue is to train on more image data, as our train set contains only 8500 images (6000 after balancing). Thus, some less frequent but offensive images can be successfully identified. A potential solution for the latter issue is to improve on the structure of our current best model.

The image and word embeddings are joined using a simple concatenation (early fusion). As discussed above, early fusion can lead to lack of homogeneity and larger prediction error. We can potentially try some more advanced fusion algorithm such as deep learning-based CNN or RNN fusions, as detailed in [18].

In addition, while external features can be incredibly helpful as shown by the large increase of accuracy of Model 3 compared to Model 2, we need to note that these tags can negatively bias the model and some tags could be wrong in first place. Therefore, further work is needed to improve the model architecture and feature space/fusion. One possible considerations for the feature space would be to leverage Web Entity Detection, as Zhu did in his winning solution [27] in order to help the model understand real-world contextual information. Another possibility is to still use the race and gender tags but this time fuse this prior to the VisualBERT pre-training so that the information gets included in the VisualBERT representation.

6. Conclusion

In this paper, we worked on classifying hateful memes using three models: a baseline model that concatenates image embeddings from Resnet152 and sentence embeddings from S-BERT and feeds the concatenation through linear and ReLU layers; a VisualBERT model that concatenates image embeddings from R-CNN and text embeddings from BERT and feeds the concatenation through a Transformer; and a VisualBERT with face feature extractions that concatenate extra racial and gender tags. The model with the highest accuracy is the VisualBERT with feature extractions, despite that it has a lower AUC potentially due to the imbalance between type I and type II errors.

As discussed above, the external features helped improve the accuracy, or, more specifically, the model's ability to correctly identify non-hateful memes at some cost of hateful meme classification accuracy. Given this result, if provided with more resource, we might consider adding other external features to further improve the accuracy and update the decision rule in our model to balance the false positive and false negative more.

References

[1] Searches of hateful speech detection. https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&as_ylo=2005&as_vis=1&q=hate+speech+detection+social+media&btnG=. Accessed May 30, 2022. 1

[2] 3 stats that show what memes mean to gen z & millennials. https://www.ypulse.com/article/2019/03/05/3-stats-that-show-what-memes-mean-to-gen-z-millennials/, March 2019. 1

[3] Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *arXiv:1705.09406v2 [cs.LG]*, 2017. 2, 3

[4] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013. 5

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805 [cs.CL]*, 2018. 2, 3, 4

[6] ASEY FITZPATRICK. How to build a multimodal deep learning model to detect hateful memes. https://www.drivendata.co/blog/hateful-memes-benchmark/, June 2020. 6

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv:1512.03385 [cs.CV]*, December 2015. 2

[8] Constance Iloh. Do it for the culture: The case for memes in qualitative research. *International Journal of Qualitative Methods*, January 2021. 1

[9] William C Sleeman IV, Rishabh Kapoor, and Preetam Ghosh. Multimodal classification: Current landscape, taxonomy and future directions. *arXiv:2109.09020v1 [cs.LG]*, 2021. 2, 3

[10] Kimmo Karkkainen and Jungseock Joo. Fairface: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1548–1558, 2021. 4, 5

[11] Douwe Kiela, Suvrat Bhooshan, Hamed Firooz, Ethan Perez, and Davide Testuggine. Supervised multimodal bitransformers for classifying images and text. *arXiv:1909.02950 [cs.CL]*, September 2019. 2

[12] 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, 2, 4

[13] Luinian Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv:1908.03557 [cs.CV]*, 2019. 2, 3, 4

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context. *arXiv:10.48550/ARXIV.1405.0312*, May 2014. 2

[15] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv:1908.02265 [cs.CV]*, August 2019. 2

[16] Jitendra Singh Malik, Guansong Pang, and Anton van den Hengel. Deep learning for hate speech detection: A comparative study. *arXiv:2202.09517[cs.CL]*, 2022. 1

[17] Niklas Muennighoff. Vilio: State-of-the-art visio-linguistic models applied to hateful memes. *arXiv:2012.07788 [cs.AI]*, December 2020. 2

[18] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. Multimodal deep learning. In *ICML*, pages 689–696, 2011. 3, 8

[19] Esteban Ortiz-Ospina. The rise of social media. https://ourworldindata.org/rise-of-social-media, September 2019. 1

[20] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 3

[21] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv:1908.10084*, 2019. 1, 3

[22] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv:1506.01497 [cs.CV]*, June 2015. 2

[23] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. pages 2556–2565, July 2018. 2

[24] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv:1908.08530 [cs.CV]*, August 2019. 2

[25] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, Oct. 2020. Association for Computational Linguistics. 4

[26] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 4

[27] Rong Zhu. Enhance multimodal transformer with external label and in-domain pretrain: Hateful meme challenge winning solution. *arXiv:2012.08290v1 [cs.CL]*, 2020. 1, 2, 4, 8