Hateful Memes Detection

Christopher Bravo
cbravo8@gatech.edu

Tejas Lokeshrao
tejaslkr23@gatech.edu

Viktorya Poghosyan
viktoryapoghosyan@gatech.edu

William Watson
wwatson43@gatech.edu

Abstract

The goal of this project is to detect hateful content in memes by leveraging several transformer-based models. This requires an understanding of the text and image aspects of the meme, so our solution was to deploy a series of early fusion and intermediate fusion models. The intermediate fusion models saw more success as the embeddings in the model contained both the text and image vectors. We leveraged the HateCLIPper architecture as our base model, and hyper-tuned it and created variations to feed into an ensemble learning model to detect hatefulness. This resulted in an AUROC of 0.865, which is better than what was presented in the original HateCLIPper paper.

1. Introduction/Background

One issue in the modern internet age is how to identify hateful or offensive content. Content on the internet (e.g. memes) often employs both text and image modalities, which adds complexity in determining whether content is hateful. Though the individual text and image modalities may be harmless in their own capacity, when combined they can form offensive innuendos. For this project our team deployed numerous versions of unimodal and multimodals models and compared their accuracy in assessing whether memes are hateful. Our main goal was to determine which model and parameters are most accurate for classifying hateful memes.

In order to solve the issue of categorizing multi-modal content three main categories of methodologies have historically been employed: early, intermediate, and late fusion models. [8] Early fusion models combine the image and text modalities and feed the concatenated result into a classifier in order to classify the multimodal content. This approach enables detection of harmful content when the individual text and image modalities are not offensive in and of themselves. Intermediate fusion models are similar in structure to early fusion models but instead only combine modalities after they have undergone some level of processing, such as feature extraction or initial filtering. Late fusion models, unlike the earlier fusion models, classify each modality before combining the results. Each modality's classification is then fed into a final classification model to assess the entire input.

Generally, early fusion models suffer from creating overly complex feature spaces due to the complexity involved in processing multi-modal concatenated input. Intermediate fusion models, while less complex in feature space, can be difficult to tune. Often, it is difficult to assess the proper level of processing required for each modality before fusion. Late fusion models, due to their simplicity, cannot assess sentiment unless it is present in at least one of the modalities. For the purpose of assessing hateful memes, this limitation undermines detection of memes whose individual modalities are not offensive.

1.1. Motivation

If successful, this project will indicate which multimodal models and corresponding parameters are most accurate in classifying hateful content, which has several ethical and societal implications. From a social media perspective, more accurate detection models, when deployed, can serve to filter and ban illicit content. Moreover, they can act to flag individuals or accounts who are spreading hate. When applied in a national security context, these models can identify hateful language and images which may be indicative of future racially or religiously motivated acts of terror or aggression.

1.2. Dataset

Our source dataset consisted of the "Hateful Memes" dataset created by Facebook as part of their Hateful Memes competition. The competition was originally associated with a $100,000 prize and was first released in May 2020. The dataset consisted of 10,000+ multi-modal examples, meaning they contain text and images, and they are each labeled as "Hateful" or "Not Hateful". The text part of the images has also been extracted and placed in text files, so they are easier to work with. [6]

2. Approach

2.1. Data Augmentation

Since the hateful memes dataset provided by Facebook contains just over 10,000 elements, we attempted to generate additional meme images through data augmentation. By creating variations of data that was already present, this would have made the dataset more robust and allowed models to generalize better. For the text aspect, we generated paraphrased versions of the captions with the assistance of GPT 3.5. For the image aspect, we attempted using SRNet [22] in order to edit images with a style-retention network. This converted the text parts of the image into blurry segments, causing the overlapped text to not be comprehensible. We additionally attempted to detect regions from images using OCR systems, mask those regions, use generative models (DeepFill) to fill those regions, and put paraphrased text on top of it. Unfortunately, parts of the image were modified outside of the masked section. Therefore, we decided to move forward with the base dataset and develop our early and intermediate fusion models.

2.2. Early Fusion Models

The early fusion approach we deployed performed image captioning on the source-image of the meme. This image captioning along with the original text in the meme were then fed into a transformer in order to label memes as hateful. As this is a binary classification problem, BCE loss was used, as was an ADAM optimizer.

2.2.1 Image captioning models

Three different image captioning models were tried. The first was based on an implementation by DeepRNN of the paper "Show, Attend, and Tell". [23] [4] The next model employed was MPLUG [14] using the modelscope pipeline [5], the current state-of-the-art for image captioning [3]. The state-of-the-art runner-up, OFA [21], was the final selection. See section 3 for the reasons behind this selection.

As MPLUG and OFA sometimes directly picked up and removed the text in the memes, Tesseract OCR [10] was used to detect the text location, which was then in-filled using a Telea approach. [19]

For the early fusion model, the image captions were combined with the actual meme texts in a format similar to `The image is of <caption> with the text '<text>'` and then fed into a model. Results were compared across several different transformer configurations as described below.

2.2.2 Pre-trained sentence embedding transformers

First, pre-trained sentence embeddings (SBERT) were employed to generate numeric vectors representing each image caption/text combination in high-dimensional space. [17] We then added a linear classifier to the end of the model to convert these embeddings into a 0-1 score. We utilized two separate language models, all-MiniLM-L6-v2 and all-mpnet-base-v2, to process the generated image captions and meme text. Due to the increased dimensional space of all-mpnet-base-v2 (768 dimensions) compared to all-MiniLM-L6-v2 (384 dimensions) we anticipated improved accuracy in the model utilizing all-mpnet-base-v2.

We employed two variants of the linear classifier. In the first variant we had a 150-node layer, followed by a 50-to-1 layer. The smaller variant, conversely, went only to the 50-node layer and thence to 1. Both versions used ReLUs and a dropout of 0.1. By comparing the two linear layer sizes we could determine whether the added complexity in the larger linear layer could enable better classification of text representations.

2.2.3 Fine-tuning pre-trained text classifier

We also deployed a series of transformer models which utilize pre-trained text classifiers, including BERT (Bidirectional Encoder Representations from Transformers) [9], ALBERT (A Lite BERT) [12], DistilBERT (a version of BERT using knowledge distillation) [18], RoBERTA (Robustly Optimized BERT Pretraining Approach) [15], and GPT2 (Generative Pre-trained Transformer 2) [16]. Although previous unimodal approaches have utilized BERT for classifying hateful memes, our approach differs in employing the modified versions of BERT (e.g. ALBERT, DistilBERT, and RoBERTA). Moreover, little research has examined GPT-2's efficacy in evaluating hateful memes, another area our methodology differs.

Note that these transformer models each have their own unique strengths which make them worthy of comparison. Known for its deep understanding of language context, BERT excels in tasks requiring nuanced language interpretation, but its large size can be resource-intensive, potentially hindering performance in resource-limited environments. [9] ALBERT offers a more efficient alternative to BERT, preserving much of its contextual understanding capabilities while being lighter and faster, though it might trail BERT in understanding complex language patterns due to its reduced size. [12] Similarly, DistilBERT may also lag BERT due to its tendency to balance performance with efficiency. [18]

2.2.4 Zero- and few-shot prompting

We also utilized several transformers with zero or few shot prompting including Falcon-7B, Falcon-40B, Zephyr-7B, and BART. The Falcon models are a LLMs released by the TII [7], while Zephyr-7B is an LLM built by HuggingFace and is derived from Mistral. [20] BART was developed by Facebook. [13]

For zero-shot sampling, we utilized a prompt like `There is an image of <description> and a caption that says "<text>". Would you say this meme is rude, offensive, or otherwise hateful? Answer "yes" or "no" only - no other answers are permitted.` We utilized this prompt for Falcon-7B, Falcon-40B, and Zephyr-7B.

For few shot prompting, we added two additional examples to the above prompt in the form of description=`skunk` or `rose` and text=`love the way you smell today`. Few-shot prompting was tried only with Falcon-7B. Additionally, top-10 sampling was tried with the original prompt (on Falcon-7B only). BART was unique in that it is a pre-trained model built specifically for zero-shot classification – in this case we used the classes ”offensive” and ”harmless”.

The LLMs occasionally disobeyed the ”yes” or ”no” instruction, and so simple rules-based postprocessing was implemented (such as looking for the presence of a ”yes” or ”no” or synonyms, with a default classification of ”hateful”).

Falcon-7B and Zephyr-7B both use seven billion parameters and are a causal decoder-only models; both have also been fine-tuned previously for chat. [7] [13] Falcon-40B expands on Falcon-7B to feature 40 billion parameters and is trained on a trillion tokens from the Refined Web dataset.

By utilizing these specific language models we expanded on the methodologies historically employed for unimodal text processing for hateful memes (none of which have used LLMs). Moreover, by utilizing BART, we could also test the influence of bidirectional and autoregressive transformers on detecting hate speech. This bidirectional understanding we predicted would allow the model to detect the nuances present in hateful content.

Generally, we anticipated these language models to outperform the other pre-trained text classifier and sentence embedding transformers alluded to in the previous sections. We believed this outperformance would be due to the sheer size of these models along with the diversity of training data they had exposure to.

2.3. Intermediate Fusion Models

2.4. MMBT

In contrast to the early fusion approach to detecting hateful memes, we additionally implemented multimodal approachs that took both the image and caption features into account when being fed into the model. Our research indicated that this is the approach most successful models take, as memes have equally important text and image components. One such approach is the MultiModalBi-Transformer (MMBT). Its flexible architecture enables us to customize the model with different image and text encoders and fuse the embeddings to train multimodal meme classification. We used Contrastive Language-Image Pretraining (CLIP)-based image encoders and BERT-based text encoders trained to classify between hatespeech, offensive, or normal.

2.5. HateCLIPper

A variation of this approach is presented in HateCLIPper: Multimodal Hateful Meme Classification based on Cross-modal Interaction of CLIP Features (see Figure 1). Hate-CLIPper leverages CLIP encoders to align image and text representations of a meme in a joint embedding space. From this embedding space of text and image vectors, the model utilizes a Feature Interaction Matrix (FIM) to illustrate the cross-modal interactions between these features. This outputs a probability score for a meme being hateful.

As illustrated, there are trainable projection layers built into the HateCLIPper architecture. This is due to the fact that the pretraining dataset for CLIP contained image-text pairs that conveyed the same meaning, and this is not necessarily the case for memes. Therefore, the model needed to be retrained with portions of the Hateful Memes dataset to fit the data better. From their experimentation, they were able to obtain a test seen AUROC of 0.838 with 5 million trainable parameters. As these results were fairly high, we decided to maintain their architecture and hyper-tune the parameters to obtain better results.

In order to further improve results, we created an ensemble learning model that considered our tuned HateCLIPper model and another model that took a more direct approach. Instead of using the FIM to capture interactions between the text and image vectors in the image, we perform element-wise multiplication on the two vectors and feed that into the model. This varies from the FIM as it calculates pairwise interactions by performing the dot product of an image with every text feature. The hyperparameters for this model were the same as the HateCLIPper model. To create the ensemble learning architecture, we obtained the moving average of the weights from the two models, and fed that into a new model for classification.

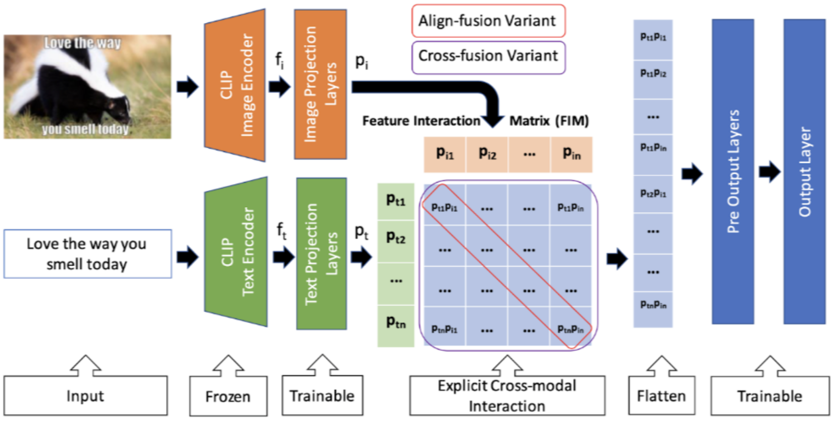

Figure 1. Architecture of HateCLIPper [[11]]

Hyperparameter	Value
Image Size	224
Pretrained CLIP model	ViT-Large-Patch336
Projection dimension (n)	1024
Pre-output dimension (m)	1024
Optimizer	AdamW
Number of pre-output layers	2
Maximum epochs	20
Batch size	64
Learning rate	0.001
Gradient clip value	0.0001
Weight decay	0.01
Map Dimensions	1024

Table 1. Tuned Hyperparameters for our model

2.6. Model Challenges

Over the course of this project there were several problems we anticipated due to the scope and size of the language models we deployed. From a training perspective, we allotted additional time for our larger models to train. In particular, we did not have access to any GPUs substantial enough to run the LLMs, so all were run on the CPU, a fairly slow process. Falcon-40B, for instance, took over two days to run on just the test set and consumed hundreds of GB of RAM. The other transformer models could fit on the 12GB GPUs we had access to, but the batch sizes had to be relatively small (2 or 4 each, and some with half-precision training in order to fit). To compensate for the potential increased volatility of this approach, lower learning rates were used.

Another challenge we anticipated was the lack of training data for explicit content. The open-source models we utilized such as BERT, Falcon, GPT-2 etc., are largely trained on sophisticated text resources that have been specifically cleaned of offensive data. We therefore predicted that these models may have issues when performing classification on explicit text they had not seen before.

Moreover, we found the open-source models at times were uncooperative in assessing the memes. Falcon-40B, for instance, consistently responded “As an AI language model, I cannot determine the intent behind the meme. However, it is important to note that humor can be subjective...” This limitation mitigated our ability to test these models with illicit content.

3. Experiments and Results

The metrics we used for measuring success were accuracy (the percentage of predictions correct) and area under the receiver operator characteristic curve (AUROC). These performance metrics were used in the Hateful Memes challenge [1] and capture both the ability of the model to correctly identify true positives and its capacity to avoid false positives. This dual focus helps in ensuring that the model is not only accurate but also reliable in differentiating between nuanced cases, an essential aspect in the context of detecting and interpreting complex social media content like memes. Moreover, usage of the same performance metrics as the challenge allowed for comparison between our models and previously employed methods.

For reference, the humans can reach an AUROC about 0.826. [11]

3.1. Early fusion

All portions of these approaches were done in PyTorch, either directly (sentence embeddings) or via modelscope [5] or HuggingFace [2] pipelines. The exception is the ”Show, Attend, and Tell” model [4] which used TensorFlow.

Understanding which memes are hateful requires a deep and nuanced understanding of the world in general. To capture this requires vast amounts of varied training data, and so the use of pretrained models as starting points allows us

Model	Accuracy	AUROC (if available)
all-MiniLM-L6-v2 sentence embedding + smaller classification head	59.71%	0.60
all-MiniLM-L6-v2 sentence embedding + larger classification head	60.77%	0.63
all-mpnet-base-v2 sentence embedding + smaller classification head	61.15%	0.64
all-mpnet-base-v2 sentence embedding + larger classification head	63.37%	0.66
ALBERT	57.02%	0.5
DistilBERT	63.85%	0.61
RoBERTA	57.02%	0.5
BERT	57.02%	0.5
GPT-2	65.9%	0.65
Falcon-7B zero shot	57.02%	-
Falcon-7B sampling	44.9%	-
Falcon-7B few shot	57.02%	-
Zephyr-7B zero shot	55.5%	-
Falcon-40B zero shot	N/A	-
BART	44.6%	0.39

Table 2. Early fusion results

to leverage this information in a way that would otherwise be completely infeasible, while the fine-tuning allows us to focus this already-learned knowledge to the task at hand.

3.1.1 Image captioning

The image captioning models were evaluated qualitatively. The "Show, Attend, and Tell" model was by far the worst, with captions such as "a man with a hat and a hat". The MPLUG model produced more coherent captions, but occasionally generated only the caption "a". This premature EOS token could be due to corruptions in the training data or a total lack of representation of the offending images in the training set (and therefore the model can make no better guess than the description starting with an article). The OFA model, however, performed well on most images with no "a" problem.

3.1.2 Prediction results

The best results for every early fusion approach attempted are shown in Table 3. Note that the LLMs do not produce a score and so a ROC curve cannot be generated for them; Falcon-40B also refused to answer any question and so could not be evaluated.

The sentence transformer models were trained to determine how model size influences the prediction outcomes, as well as to determine how well a pretrained embedding model generalizes to a new task. In general, the larger models (both those used in the pretrained embeddings and the classification head) performed better (see Figure 4). This makes sense as the larger pretrained embeddings can contain more information about a given sentence, and so can better capture the nuance of language.

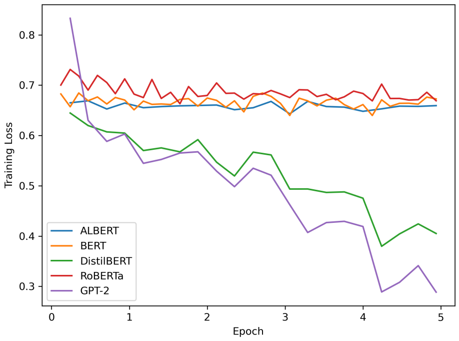

Figure 2. Fine-tuning transformer loss

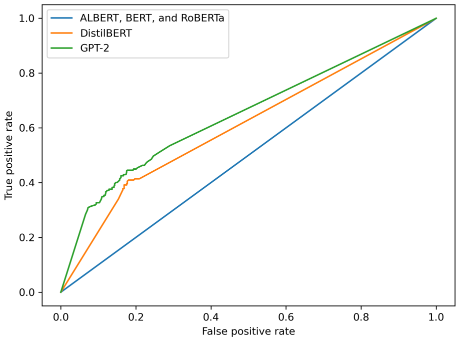

Figure 3. Fine-tuning transformers ROC

The fine-tuned transformer models were compared with the various few- and zero-shot approaches, and in general they performed better. Several of the transformers, including ALBERT, BERT, and RoBERTA, failed to train at all (see Figure 2). This is likely due to the tiny batch sizes required by GPU limitations resulting in too much noise for a stable training. On the other hand, the knowledge distillation in DistilBERT enabled it to be smaller (and thus a larger batch size could be used) while still maintaining most of the performance of the original model. GPT-2 has different architecture (causal rather than bidirectional attention) than the BERT-based models and also used more training data, which maybe have contributed to its success (see Figure 3). The DistilBERT and GPT-2 models began to overfit after about 5-6 epochs, and so training was stopped then. The best fine-tuned transformer models also outperformed the sentence embedding models, likely because they were specifically trained for this task, while the embeddings are more generic and were ultimately designed for determining how similar two sentences are.

Counter to our expectations, the zero/few-shot approach performed the worst, with only one approach (Zephyr-7B) doing better than chance. Falcon-7B with zero shot prompt-

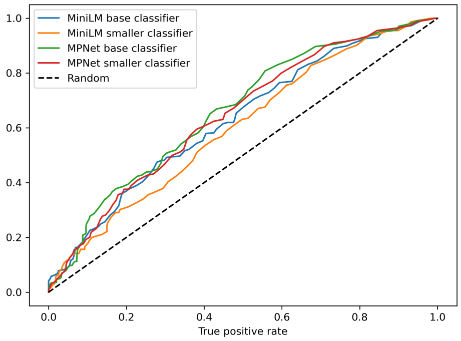

Figure 4. Sentence captioning ROC

Model	Accuracy (if available)	AUROC (if available)
Resnet50x4 + bert-base-uncased-hatexplain	57.81%	0.73
Resnet50X16 + bert-base-uncased-hatexplain	57.87%	0.74
ViT-B/32 + bert-base-uncased-hatexplain	58.92%	0.76
ViT-B/32 + bert-base-uncased-hatexplain-rationale-two	57.93%	0.745
HateCLIPper (default)	75.7%	0.858
HateCLIPper-FIM (tuned hyperparameters)	76.01%	0.840
HateCLIPper-Non-FIM (tuned hyperparameters)	76.24%	0.864
Ensemble Learning of Best HateCLIPper models	76.26%	0.865

Table 3. Intermediate fusion results

ing answered ”yes” to every meme being hateful, while Falcon-40B refused to give an answer every time. These results could be attributed to the smaller number of parameters in the model (as compared with the hundreds of billions/trillions used in the largest models), as well as the fact that these are built for generation/token prediction, rather than sentiment analysis (which is simply a byproduct of that function). Additionally, most LLMs are trained on data specifically cleaned of hateful content, thereby limiting the training exposure of these models to exactly the kind of material we wish to analyze. Additionally, the use of top-10 sampling in Falcon-7B resulted in much more variability in the output, including some grammatically correct but incoherent results - this is a further limitation of these kinds of models when compared with those specifically built and trained for the task. The difference between Falcon and Zephyr is likely due to their specific architectural differences, along with the training data used.

3.2. Intermediate fusion

3.2.1 MMBT

Performance of the MMBT model was evaluated across several encoder configurations. Ultimately, the larger image encoders (specifically ViT-B/32) performed best (AUC = 0.76), which suggests the model’s ability to understand and interpret complex visual data is key to its overall effectiveness in evaluating hateful memes. Given the complexity of memes and the nuance of images online, it is reasonable that more complex image encoders would improve results.

3.2.2 HateCLIPper

When evaluating the results of the HateCLIPper model, we also determined that large image encoders coincide with improved results. Our implementation of HateCLIPper, which utilized `clip-vit-large-patch14-336` and an ensemble approach, achieved an AUROC of 0.865, whereas the original HateCLIPper had an AUROC of 0.858.

The improved performance of our modified HateCLIPper model can also be attributed to an interpretation layer we added to the model in order to learn complex features from the fused modalities. We also adjusted the learning rate to 0.001 to 0.0001 to help the model converge to a better solution and increased the dropout to mitigate overfitting.

The enhanced performance of the ensemble approach can likely be attributed to the voting mechanism between the sub-models in the ensemble, which enabled them to capture the subjectivity of assessing whether a meme is hateful.

4. Conclusion

Overall, intermediate fusion outperformed the early fusion. This is likely because they more directly combine the different meme modalities. The early fusion models also rely on a successful image captioning model; while ours was good, it did fail to capture the nuance in some images and was likely the weakest link in the early fusion chain.

As our ensemble learning approach had the best performance, the next steps in advancing our hateful meme detection model involve refining it. To enhance its capabilities, fine-tuning on more diverse datasets containing explicit content is essential so the model encounters a broader spectrum of hateful memes during training. A careful analysis of misclassified images would provide valuable insights for targeted adjustments, addressing specific challenges in the recognition process. Additionally, ongoing improvement necessitates a continuous feed of new, hateful content to the model, allowing it to adapt and evolve with emerging trends. This iterative approach ensures the model’s responsiveness to evolving online content dynamics, contributing to its sustained accuracy and effectiveness in identifying and combating offensive material.

Student Name	Contributed Aspects	Details
Viktorya Poghosyan	Intermediate Fusion (MMBT Models and HateCLIPper Optimization)	Implemented intermediate fusion models with MMBT, trained and did experiments with multiple image/text encoders. Also used HateCLIPper code base to train models with various configurations, and implemented ensembling mechanisms. Contributed to comparative analysis of models. Analyzed the effect of hyperparameter changes.
William Watson	Early Fusion (Image Captioning, Sentence Embedding Pipeline, Transformers and LLMs)	Experimented with different image captioning models. Additionally built the sentence embedding pipeline and trained and experimented with it. Contributed to fine-tuning and comparative analysis of transformer models and LLMs.
Christopher Bravo	Transformers and LLMs	Implemented and experimented with fine-tuning transformer models. Ran and experimented with pre-trained LLMs. Contributed to comparative analysis of models.
Tejas Lokeshrao	Data Augmentation and MMBT Models	Experimented with data augmentation techniques such as paraphrased text generation with GPT 3.5 API, text edition with SRNet and OCR/DeepFill combination. Helped to develop intermediate fusion models with MMBT, trained and did experiments with multiple image/text encoders. Contributed to comparative analysis of models.

Table 4. Contributions of team members.

References

[1] Hateful memes challenge winners. https://ai.meta.com/blog/hateful-memes-challenge-winners/. 4

[2] Hugging face. https://huggingface.co/. 4

[3] Image captioning. https://paperswithcode.com/task/image-captioning. 2

[4] image_captioning. https://github.com/DeepRNN/image_captioning. 2, 4

[5] modelscope. https://www.modelscope.cn/home. 2, 4

[6] Hateful memes challenge and dataset for research on harmful multimodal content. https://ai.meta.com/blog/hateful-memes-challenge-and-data-set, May 2020. 2

[7] Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Merouane Debbah, Etienne Goffinet, Daniel Heslow, Julien Launay, Quentin Malartic, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. Falcon-40B: an open large language model with state-of-the-art performance. 2023. 3

[8] Said Yacine Boulahia, Abdenour Amamra, Mohamed Ridha Madi, and Said Daikh. Early, intermediate and late fusion strategies for robust deep learning-based multimodal action recognition. *Machine Vision and Applications*, 32(121), 2021. 1

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. 2

[10] Anthony Kay. Tesseract: An open-source optical character recognition engine. *Linux J.*, 2007(159):2, jul 2007. 2

[11] Gokul Karthik Kumar and Karthik Nandakumar. Hate-clipper: Multimodal hateful meme classification based on cross-modal interaction of clip features, 2022. 4

[12] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations, 2020. 2

[13] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *CoRR*, abs/1910.13461, 2019. 3

[14] Chenliang Li, Haiyang Xu, Junfeng Tian, Wei Wang, Ming Yan, Bin Bi, Jiabo Ye, Hehong Chen, Guohai Xu, Zheng Cao, Ji Zhang, Songfang Huang, Fei Huang, Jingren Zhou, and Luo Si. mplug: Effective and efficient vision-language learning by cross-modal skip-connections, 2022. 2

[15] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019. 2

[16] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. 2

[17] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. 2

[18] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2020. 2

[19] Alexandru Telea. An image inpainting technique based on the fast marching method. *Journal of Graphics Tools*, 9(1):23–34, 2004. 2

[20] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. Zephyr: Direct distillation of lm alignment, 2023. 3

[21] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. *CoRR*, abs/2202.03052, 2022. 2

[22] Liang Wu, Chengquan Zhang, Jiaming Liu, Junyu Han, Jingtuo Liu, Errui Ding, and Xiang Bai. Editing text in the wild. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, page 1500–1508, New York, NY, USA, 2019. Association for Computing Machinery. 2

[23] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2048–2057, Lille, France, 07–09 Jul 2015. PMLR. 2