VILIO: STATE-OF-THE-ART VISIO-LINGUISTIC MODELS APPLIED TO HATEFUL MEMES

Niklas Muennighoff
Peking University
https://github.com/Muennighoff
muennighoff@stu.pku.edu.cn

ABSTRACT

This work presents Vilio, an implementation of state-of-the-art visio-linguistic models and their application to the Hateful Memes Dataset. The implemented models have been fitted into a uniform code-base and altered to yield better performance. The goal of Vilio is to provide a user-friendly starting point for any visio-linguistic problem. An ensemble of 5 different V+L models implemented in Vilio achieves 2nd place in the Hateful Memes Challenge out of 3,300 participants. The code is available at `https://github.com/Muennighoff/vilio`.

1 Introduction

Language always appears as part of a different medium, such as written text in vision or spoken text in sound. For some problems, extracting the language only suffices for useful machine learning models. Pure language models have therefore had considerable impact on our lives in areas such as machine translation, text classification tasks or language reasoning problems [1]. Often, however, the underlying medium is relevant and must be understood in conjunction with the language.
One such area is Internet memes. They combine an image and a superimposed text to provide a nuanced message. Often the image or text on its own is not enough to convey the message. That nuanced message can be hateful, but manual classification and removal of hateful memes is costly for social platforms. The Hateful Memes Challenge [2] proposed by Facebook AI and hosted on the DrivenData platform aims to leverage machine learning models to solve this problem.
Current state-of-the-art Vision+Language machine learning models are based on the transformer architecture [3]. Among them, there are two prevalent approaches: **Single-stream models**, such as VisualBERT [4], UNITER [5], OSCAR [6], use a single transformer to process the image and language input at the same time. **Dual-stream models**, such as LXMERT [7], ERNIE-ViL [8], DeVLBERT [9], VilBERT [10], rely on separate transformers for vision and language, which are then combined towards the end of the model.
This work contributes the following:

- A code base of 12 different vision+language models organized like huggingfaces transformers library [11] to promote future V+L research
- Performance-enhancing modifications applied to state-of-the-art V+L models
- An evaluation of different models on the Hateful Memes Dataset [2] & necessary code for future research on the dataset

2 Problem Statement

The Hateful Memes Dataset [2] consists of a training set of 8500 images, a dev set of 500 images & a test set of 1000 images. The meme text is present on the images, but also provided in additional jsonl files. To increase its difficulty, the dataset includes text- & vision confounders. Such confounders change from being hateful to non-hateful or vice-versa by swapping either text or image only. Figure 1 is one such example. They ensure that models must reason about both vision & language. A vision-only or language-only model cannot succeed in the task.

(a) Hateful meme

(b) Non-hateful meme

Figure 1: Predicted RoIs & the top four predicted objects on a confounder example from the Hateful Memes Dataset. The hateful meme has been neutralized by removing & replacing E.T. with an unrelated image. ©*Getty Images*

Participants in the competition submitted a probability for the likelihood of a meme being hateful for each of the 1000 test images. The submitted probabilities were then used to calculate the area under the curve of the receiver operating characteristic:

$$AUROC = \int_{x=0}^{1} \text{TPR}(\text{FPR}^{-1}(x))\, dx \tag{1}$$

The AUROC score was used as the competitions key metric. Intuitively, it penalizes models that are bad at ordering memes by hatefulness. Whether the probability values itself are high or low does not matter, only how they are ranked.

3 Approach

A high-level overview is present in Figure 2. The pipeline consisted of three stages: Preparation, Modelling & Ensembling.

3.1 Preparation

In a first step, features were extracted from images using the detectron2 framework [12]. Using features instead of whole images speeds up the training process and still captures the most relevant content. It has been shown that using diverse features can boost performance [13]. Therefore, different models are used for feature extraction, pre-trained on VisualGenome with & without attributes [14]. In addition, varying Regions of Interest (RoIs) are kept in each feature extraction. Min- & Maxboxes are set to the same number either 36, 50, 72 or 100. Figure 1 shows an example of predicted RoI's and the four most commonly predicted objects on two memes.
Together with the meme text, which has been extracted using optical character recognition (OCR) and provided in the dataset, features are then fed into the models.

3.2 Modelling

The following provides an overview of general settings and changes made to individual models that were used in the final solution to the Hateful Memes Challenge. The code repository also provides additional models that were not used in the submission.

3.2.1 General Settings

Pre-trained weights provided by the original authors of the V+L models are used. VisualBERT & OSCAR are also task-specific pre-trained on the Hateful Memes Dataset. All models are then finetuned using binary cross-entropy loss and a batch size of 8. The Adam optimizer [15] is used with a learning rate of 1e-5 and 10% linear warm-up steps [16]. Gradients are clipped at 5 for VisualBERT, OSCAR & UNITER and at 1 for ERNIE-ViL. VisualBERT, OSCAR & UNITER are trained for 5 epochs and Stochastic Weight Averaging [17] is used during the last 25% of training. ERNIE-ViL models are trained for 5000 steps. The weights from the last step are taken for all models and used for inference on the test set.
Overall, not much time was spent on hyperparameter optimization, as fundamental architecture changes have a larger impact.

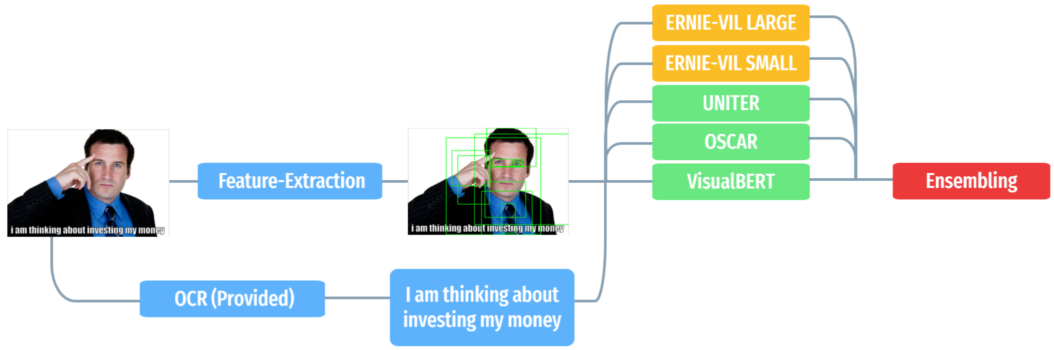

Figure 2: Pipeline on the Hateful Memes Dataset using Vilio split into three stages: Preparation, Modelling & Ensembling ©*Getty Images*

3.2.2 ERNIE-ViL

The ERNIE-ViL model by PaddlePaddle [8] is based on VilBERT [10] and the ERNIE transformer [18]. In addition to extracted features, the original ERNIE-ViL is pre-trained on ground-truth labelled boxes. As there are no ground-truth boxes provided for the Hateful Memes Dataset, a second set of features extracted with 10-100 boxes is used as a fake ground-truth. In addition to pre-trained weights on Conceptual Captions (CC) [19], task-specific pre-trained weights on VCR [20] are used to increase diversity.

3.2.3 UNITER & OSCAR

Updates are made to both UNITER & OSCAR to reflect changes in the transformers library [11], such as an updated activation function and embedding calculations. OSCAR is also task-specific pre-trained on Hateful Memes using Image-Text-Matching (ITM) and Masked Language Modelling (MLM). Adding the predicted objects & attributes during task-specific pre-training as in the original implementation was not beneficial. The classifier from LXMERT with GeLU activation [7] is used. The OSCAR & UNITER pre-trained weights are based on the BERT transformer [21]. The code to use RoBERTa [22] and other language transformers is provided in Vilio. However, without the pre-training, they cannot match performance.

3.2.4 VisualBERT

Similar to UNITER & OSCAR, VisualBERT was updated based on the transformers library as of September, 2020. Like OSCAR, VisualBERT model is task-specific pre-trained using MLM. While the original VisualBERT uses the same token type ids for the language & visual input, the Vilio implementation creates a separate visual token type. Therefore, token type weights are reinitialized and retrained from scratch. This improved the model by an absolute 1.2% on the AUROC metric. Multi-sample dropout [23] and learnt weights for averaging of the transformer layers further improve the model. A linear classification head is used with 500 times the learning rate of the transformer backbone.

3.3 Ensembling

For each of the models 3-5 seeds with different extracted features are averaged. Subsequently, the averaged predictions of each model are fed into an ensembling loop that applies Simple Averaging, Rank Averaging, Power Averaging & Simplex Optimization [24] to produce a final submission. The weights for the Simplex Optimization are learned based on dev set predictions and then applied to test set predictions trained on the full dataset (train+dev).

4 Results & Discussion

4.1 Performance & Limits

The individual models performance on the validation set and test set can be seen in Table 1. When ensembled, Vilio effectively closes the gap between baseline models & human performance on Hateful Memes. The organizers have noted, however, that the humans tested on were not experts and that the actual human performance might be closer to 100%. Investigating this and establishing a new human baseline makes for an interesting future direction.
While not reported, the ensemble accuracy of Vilio using absolute labels, not probabilities, is only 75.40% compared to

Source	Model	**Validation** AUROC	**Test** AUROC
Hateful Memes Baseline	Human	-	82.65
	ViLBERT	71.13	70.45
	VisualBERT	70.60	71.33
	ViLBERT CC	70.07	70.03
	VisualBERT COCO	73.97	71.41
Vilio	VisualBERT	75.49	75.75
	OSCAR	77.16	77.30
	UNITER	77.75	78.65
	ERNIE-ViL Base	78.18	77.02
	ERNIE-ViL Large	78.76	80.59
	Ensemble	**81.56**	**82.52**

Table 1: Model performance.

84.70% for humans on the test set [2]. The reason for this is that Vilio has been optimized for ranking, but humans are better at binary predictions than probabilities. Accuracy leaves for a considerably large gap to close.
Including the averaged seeds, the Vilio ensemble is made up of 19 trained models. Especially the ERNIE-ViL models tend to be instable, hence five seeds are averaged. Adding Stochastic Weight Averaging [17] for ERNIE-ViL, not just the other models, could help tackle this issue. Achieving the same results with less seeds and less compute is something worth looking into.

4.2 Future Work

With the framework laid down, directions for future work are vast. Apart from the suggestion in Section 4.1, four ideas are:

- Can we solve hateful GIFs?
- In the Hateful Memes dataset meme captions were standardized, but in reality they may vary in font and size. This is useful information that may help the model determine hatefulness. Can we therefore integrate the OCR algorithm into the trainable model?
- With recent advances in applying transformers to vision [25], can we skip the feature extraction and apply single-stream or dual-stream transformers from scratch?
- Single-stream encoders, such as VisualBERT, seem to outperform dual-stream encoders, such as VilBERT, with the exception of ERNIE-ViL. ERNIE-ViL, which copies VilBERT's encoders, differentiates itself with changes like its Scene Graph Parser [8]. Could an ERNIE-VisualBERT model with a single-stream encoder outperform the dual-stream ERNIE-ViL?

5 Acknowledgements

I would like to thank Greg Lipstein, Casey Fitzpatrick & the others from DrivenData for hosting the competition and their patience. I also want to thank Facebook AI, especially Douwe Kiela, Hamed Firooz & Tony Nelli, for making this competition possible in the first place. Lastly, I'm very grateful for the insightful discussions I had with Liunian Harold Li, Hao Tan & Antonio Mendoza. Without your help, it would have been a lot more difficult. Thanks a lot!

References

[1] TB Brown, B Mann, N Ryder, M Subbiah, J Kaplan, P Dhariwal, A Neelakantan, P Shyam, G Sastry, A Askell, et al. Language models are few-shot learners. arxiv 2020. *arXiv preprint arXiv:2005.14165*, 4.

[2] 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.

[3] A Vaswani, N Shazeer, N Parmar, J Uszkoreit, L Jones, AN Gomez, L Kaiser, and I Polosukhin. Attention is all you need. arxiv 2017. *arXiv preprint arXiv:1706.03762*, 2017.

[4] 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.

[5] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Learning universal image-text representations. *arXiv preprint arXiv:1909.11740*, 2019.

[6] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020.

[7] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019.

[8] Fei Yu, Jiji Tang, Weichong Yin, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. Ernie-vil: Knowledge enhanced vision-language representations through scene graph. *arXiv preprint arXiv:2006.16934*, 2020.

[9] Shengyu Zhang, Tan Jiang, Tan Wang, Kun Kuang, Zhou Zhao, Jianke Zhu, Jin Yu, Hongxia Yang, and Fei Wu. Devlbert: Learning deconfounded visio-linguistic representations. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4373–4382, 2020.

[10] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, pages 13–23, 2019.

[11] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, pages arXiv–1910, 2019.

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

[13] Yu Jiang, Vivek Natarajan, Xinlei Chen, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. Pythia v0. 1: the winning entry to the vqa challenge 2018. *arXiv preprint arXiv:1807.09956*, 2018.

[14] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, 2018.

[15] Diederik P Kingma and J Adam Ba. A method for stochastic optimization. arxiv 2014. *arXiv preprint arXiv:1412.6980*, 434, 2019.

[16] Jerry Ma and Denis Yarats. On the adequacy of untuned warmup for adaptive optimization. *arXiv preprint arXiv:1910.04209*, 2019.

[17] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

[18] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. Ernie: Enhanced language representation with informative entities. *arXiv preprint arXiv:1905.07129*, 2019.

[19] 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.

[20] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6720–6731, 2019.

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[22] 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. *arXiv preprint arXiv:1907.11692*, 2019.

[23] Hiroshi Inoue. Multi-sample dropout for accelerated training and better generalization. *arXiv preprint arXiv:1905.09788*, 2019.

[24] Yong Wu, L Ozdamar, and Arun Kumar. Triopt: a triangulation-based partitioning algorithm for global optimization. *Journal of computational and applied mathematics*, 177(1):35–53, 2005.

[25] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.