Multimodal Detection of Hateful Messages Using Visual-Linguistic Pre-Trained Deep Learning Models

Yuyang Chen (ychen@putnamscience.org)
Putnam Science Academy, 18 Maple St, Putnam, CT, USA

Feng Pan
Department of Radiology, Union Hospital, Tongji Medical College, Huazhong University of Science and Technology, Jiefang Ave 1277, Wuhan 430022, China https://orcid.org/0000-0003-2820-768X

Method Article

Keywords: Deep Neural Network, Deep Learning, Active Learning, Artificial Intelligence, Hate Speech

Posted Date: March 7th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1414253/v1

Multimodal Detection of Hateful Messages Using Visual-Linguistic Pre-Trained Deep Learning Models

Yuyang Chen[1] Feng Pan[2]

ychen@putnamscience.org uh_fengpan@outlook.com

[1]Putnam Science Academy, 18 Maple St, Putnam, CT, USA; [2]Department of Radiology, Union Hospital, Tongji Medical College, Huazhong University of Science and Technology, Jiefang Ave 1277, Wuhan 430022, China.

Abstract

Online hateful messages, more commonly known as "hate speech", have recently become a major social issue. Many studies have shown them to be detrimental for both the individuals and the society. Many online platforms have employed legions of moderators to manually identify and remove these messages, yet such practices are time-consuming, expensive, and often causing mental illness among the reviewers. As a solution, computational methods are applied to automatically identify and remove hateful messages. However, as online discussions are now often dominated by memes, a format that leverages both text and image to express users' intents, many textual moderation methods have become obsolete. In order to effectively detect a hateful meme, the algorithm must possess strong vision and language fusion capability. In this work, we move closer to this goal by compositely using a Visual-Language Pre-Trained Model, an object detection model and a random forest classifier to achieve a 0.77 AUROC score on the hateful meme detection task, an improvement of 0.15 compared to the best baseline method.

Keywords— Deep Neural Network, Deep Learning, Active Learning, Artificial Intelligence, Hate Speech

Contents

1 Introduction

Hateful message, more commonly known as hate speech, has unfortunately almost become a ubiquitous phenomenon on social media. Hateful messages are often defined as content that expresses hatred against an individual or a group due to their protected characteristics. It is detrimental to both individuals and the society [1, 2, 3, 4, 5, 6]. Earlier studies found that racial hateful messages could lead to depreciation of minority's abilities [1], alienation of minorities from larger community [2], degrade in mental health, and rise in the suicide rate among minorities [3]. More recently, it was suggested that hateful messages influence the frequency of offline hate crimes [4] and produce discriminatory practices when allocating public resources [5]. Prejudice against individuals was also shown by a 2018 study to increase when the subject was frequently exposed to hateful messages [6].

Though the U.S. government is legally prohibited by the first amendment from restricting hateful messages [7], as analyzed by the American Bar Association, privately-owned digital platforms [8, 9, 10] still have unparalleled agency in taking a more active stance against hateful messages. According to the Facebook community standard, hateful message is defined as follows: "A direct or indirect attack on people based on characteristics, including ethnicity, race, nationality, immigration status, religion, caste, sex, gender identity, sexual orientation, and disability or disease. We define attack as violent or dehumanizing (comparing people to non-human things, e.g. animals) speech, statements of inferiority, and calls for exclusion or segregation. Mocking hate crime is also considered hate speech."[1] Most major digital social media platforms, including Facebook, Twitter, and YouTube, have developed respective moderation policies against hateful messages. However, current manual moderation is considered insufficient, as human moderators are slow and expensive. Besides, content moderators are known to suffer PTSD-like syndromes after repetitively reviewing violent and exploitative content [11]. As a result, attempts to automate the identification-removal process have been carried out, reducing cost while increasing speed. Most companies choose to use block-word-list to directly filter out hateful messages; however, this method is ineffective, as hateful users can use normal expressions to circumvent detection; for example, the sentence "OK Korea - you know your duty in the impending 'blackification' of the globe? I know where I stand" expresses discrimination towards black people without using hateful vocabularies, hinting that black immigration is harmful. At the same time, dictionary-based methods could cause false-positive errors, especially when historically marginalized groups are trying to reclaim the derogatory words by frequently using them in their daily communication; for example, the sentence "My nig** is on fire" can be used as a compliment in African American community. Moreover, these methods cannot be applied to any visual message, completely ignoring a substantial proportion of online speech, let alone those messages involving visual features and textual features at the same time. It is therefore necessary to develop a detection algorithm to capture the complexity beneath human expression. It is also very likely that the hateful users intentionally publish content in which neither text nor image alone can determine whether this piece of content is hateful. Therefore, a competent hateful message detection model must not only be able to process each single modality but also be capable of fusing the two modality together. Researchers try to address this challenge by introducing machine learning methods that can learn from moderation data.

[1] https://www.facebook.com/communitystandards/hate_speech

1.1 Machine learning in automatic hateful message detection

Traditionally, researchers mainly focused on detecting textual hateful messages. As early as 2009, there had been a study on detecting online harassment using a support vector machine (SVM), yet its detection accuracy was only 0.44 [12]. Over time, the detection accuracy has been improved by later studies. For example, one study had reached a modest F-score of 0.7 by first separating objective and subjective sentences in online sources, and then building the lexicon rule by extracting the word features that make the sentences subjective [13]. Another study had used pure sentiment analysis techniques to determine the category of each sentence, reaching an accuracy of 0.6 [14]. Nobata et al. studied the possibility of combining several features to detect hateful messages and reached a satisfactory f1-score of 0.78 [15]. Interestingly, another study [16]focused on the characters-side of hateful message detection and achieved a similar f1-scores of 0.79 by using SVMs with features including n-grams. Since 2009, different machine learning-based methods have been used to achieve automatic hateful message detection. Their detection accuracy was restricted due to their inability to understand the subtlety behind human language. Many hateful expressions that do not explicitly involve slurs are prone to be classified as hateful.

1.2 Deep learning in automatic hateful message detection

Another wave of studies took place since 2016, after the popularization of deep learning. Badjatiya et al. combined random embeddings, long short term memories (LSTM) and gradient boosted decision tree (GBDT) to achieve an f1-score of 0.93, 0.13 higher than their best-performing baseline model [17]. Later, using convolutional neural network (CNN) and gated recurrent unit (GRU), Zhang et al. were able to achieve an f1-score of 0.94 over the dataset provided by Davidson [18]. Some researchers even revealed the possibility of multilingual detection. Chun Ming Lee had achieved a final f-1 score of 0.95 in the Jigsaw Multilingual Toxic Comment Classification [19], a task much similar to our topic, by fine-tuning an ensemble of pre-trained language models. Though their works had largely surpassed traditional machine learning methods, they focused on only one modality; so their work could not be applied to real world situation where multimodal hateful messages are rampant.

In response, researchers have proposed multiple models that can simultaneously analyze the text and the image. In 2016, Zhong et al. studied multimodal detection of cyberbullying on Instagram. They found that by combining convolutional neural network, offensiveness score and bags-of-words, the detection algorithm can achieve a F-1 score of 0.58. Similarly, Gaumez et al. [20] proposed combining different models to jointly analyze textual and visual information for hate speech detection, achieving an accuracy of 0.69. However, they also showed that by processing input text alone, the same accuracy could be reached, probably due to the fact that many text captions. Similar situations have occurred in other tasks as well. For example, the model can achieve seemingly impressive performance in visual reasoning without any understanding of the visual content, as language can inadvertently impose strong priors[21]. Similarly, in VQA (Visual Question Answering)[22], simple baselines without sophisticated multimodal understanding can perform remarkably well[23]; and in multimodal machine translation [24], images were found to matter relatively little[25]. The authors also theorized that some hateful messages employ a lot of background knowledge which makes the relations between visual and textual elements they use very complex and diverse, and therefore difficult to learn by a neural network. The advent of Large-scale Pre-trained Models

(PTM) has shed hope into overcoming the aforementioned challenge. PTMs such as GPT (Generative Pre-trained Transformers) and BERT[26] (Bidirectional Encoder Representation from Transformer) have recently achieved great success in many complex natural language processing (NLP) tasks and become a milestone in the wider machine learning community. Thanks to the immensity of training data (for BERT, the pre-training corpus contains 3,300 million words)[26]) and the huge number of model parameters (the base version of BERT contains 110 million parameters while the large version of BERT contains 340 million parameters), some of these PTMs have surpassed human performance on multiple language understanding benchmarks[27] [28][29], such as GLUE[30]. PTMs are now generally used as backbone for downstream tasks, because the rich knowledge stored implicitly in the huge amount of model parameters could be leveraged by fine-tuning them for specific tasks. Following the success on language tasks, the community has proposed various PTMs for visual-language tasks, including VilBERT[31] and VLBERT[32]. They directly combine learned vector representations of both the text and image to reach state-of-the-art results in several multimodal tasks[33] [22]. Visual-Language PTMs are pre-trained using text-oriented loss for individual language tasks, image-oriented loss for individual image tasks, and cross-modal losses for vision-language fusion tasks. The language loss includes the usual Masked Language Modeling. The image loss varies from feature regression tasks, to object class prediction tasks. Most models adopt the Image-Text-Matching task for cross-modal pre-training. Some models add other pre-training tasks to complete their

multimodal knowledge. These large models are shown to be able to capture more complex information and perform better on many visual-language tasks. It has also been shown that a better object detection model that is used to encode visual information leads to better downstream tasks results. To address the problem of strong consistency between image and text that leads to detector's reliance on textual features in the previous hateful message dataset, and to test the true multimodal fusion capability of visual-language fusion models, Facebook Inc. has proposed a new benchmark on hateful meme detection. Specifically, many sentences or images that are harmless by themselves may become hateful when combined together. For example, in the left panel of **Figure 1**, the image of a skunk and the sentence that "Love the wayyou smell today" are both neutral, but the combined version of them intends to insult the viewer. The subtle references are easy for humans to understand yet difficult for machines to detect. On the other hand, by preserving one of the two modalities of an original meme and substitute the other part, the meaning of the new meme could be opposite to the original one. As shown in the right panel of **Figure 1**, the new meme now have the same caption inscribed as the left one, but the image part of the meme has been changed from a skunk to a rose, successfully reversing the implication of the meme from insulting to complimenting. For every hateful meme there is always a non-hateful alternative whose caption or image is changed from the original one. This kind of substitution is called "benign confounders", a technique similar to recent strategies of using counterfactual or contrastive examples[34][35][36].; they are presented to reduce the likelihood for a detector to rely on single modal information as demonstrated before.

Figure 1: These memes convey the same caption yet impart opposite messages as the word "smell" is being associated with different objects (skunk and rose). [37]

Figure 2: Neither the roses and cemetery in the image nor the caption expresses hatred alone, yet when combined, this meme expresses racism.[37]

In this work, we built two hateful meme detectors, a linear classifier and a random forest classifier, on the feature vectors produced by OSCAR+ (**O**bject-**S**emanti**C**s **A**ligned **R**e-training)[38][39], a Visual-Language PTM whose object detection (OD) module VinVl (**V**isual features **in** **V**ision-**L**anguage) can encode a more diverse collection of visual objects and concepts than typical OD models, extracting much richer semantics, richer visual concepts and attribute information. Compared with baseline methods, our model showed significant improvement in detection capability.

1.3 Methods

1.3.1 Data

Researchers at Facebook first reconstruct online memes by placing, without loss of meaning, meme text over a new underlying licensed image. They then hired annotators from a third-party annotation company

rather than a crowd-sourcing platform. The annotators were trained for 4 hours in recognizing hate speech. These annotators reconstructed the memes and rate their hatefulness on a scale of 1-3. Finally, for the memes that were labeled hateful, benign confounders were constructed. A dataset totaling exactly 10k memes were finally set up, categorized into hateful or non-hateful. A dev and test set constitute 5% and 10% of the data respectively, and rest serves as training set.

1.4 Detection pipeline

Two different machine learning classification algorithms, linear classifier and random forest, were used to classify if a given meme is hateful or not. **Figure 3** was used to outline the machine learning pipeline constructed in this study. For every meme in the dataset, in the pre-processing stage (**Figure 4**), it was passed through VinVl (**Figure 5**) and OCR (Optical Character Recognition) module to obtain an array of feature vectors for OSCAR+ to process, and this array of vectors were stacked into an input matrix. Then, on the training set, the first stage of training was carried out: OSCAR+ was connected to an external Fully-Connected Neural Network, or a binary linear classifier, predicting if an input matrix denotes a hateful or non-hateful meme; mini-batch gradient descent was carried out on the training set of 8500 images with batch size set to 50 and a learning rate of 0.000002; the loss function is set to Binary Cross Entropy Loss with Logits $L(x, y) = -(y \ln \sigma(x) + (1 - y) \ln (1 - \sigma(x))$. After OSCAR+ was successfully fine-tuned, it was connected to a random forest classifier. The random forest was further optimized on OSCAR+ output vectors, consisting of ten decision trees whose maximum depth were set to 10. The trained random forest classifier serves as the final classifier for recognizing hateful memes.

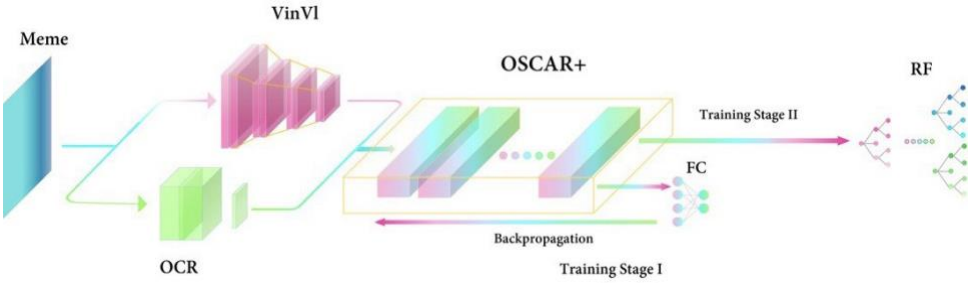

Figure 3: Full Pipeline. FC denotes Fully Connected Layers; and RF denotes Random Forest.

1.4.1 Meme Preprocessing

Because the meme itself could not be fed directly into the Visual-Language PTM, it must be preprocessed into a suitable data format. To achieve this, the meme is first input into VinVl object detector (**Figure 5**)[40] which uses the ResNeXt 152-C4[41] Convolutional Neural Network as the backbone feature extractor. In VinVl, the input meme is transformed into a feature map by the backbone network. An Region Proposal Network (RPN) is used to outline rectangular regions on the feature maps that may contain predefined categories of objects. Those features of regions of interests (ROI) are pooled into ROI-Pooling vectors of 2048 dimensions. Those vectors are then passed through different Fully-Connected (FC) Neural Networks (NN). The first FCNN is used to predict the position and the size of the bounding box for each ROI-Pooling vector. Each of these Bounding-Box-Regression vector is than concatenated with the corresponding ROI- Pooling vector to form a intermediate vector of dimension 2054. These intermediate vectors are further passed through FCNN to produce Image Feature Embedding vectors. At the same time, each ROI- Pooling vector is also passed through another FCNN that predicts the category of the object in the region corresponding to this vector. Each category corresponds to an object tag. In general, the VinVl object detector will produce a sequence of object tag embeddings and a sequence of image feature embeddings. The first sequence denotes the detected objects in a meme, the names of these objects are transformedby the OSCAR+ tokenizer into vectors of size 768. Assume that there are n objects detected in the meme, the final number of vectors will be n as well. Each of the image feature vector in the second sequence corresponds to an object tag vector in the first sequence, as they are the features extracted by the Faster-RCNN of the regions corresponding to the objects. Similarly, these image feature embeddings are of dimension 768, and there are n image feature embeddings.

On the other hand, the OCR is used to read all the text captions that appear in the meme. Assume that there are m individual words output by the OCR module, each word is transformed by the OSCAR+ tokenizer into a vector of dimension 768, and the total number of text embeddings would be $m\times$ 768. The text embeddings, the object tag embeddings, and the image feature embeddings are then further arranged in sequence, inserting between which the embedding of special token **[SEP]** that denotes different sections of OSCAR+ input. An additional embedding of the special token **[CLS]** is then appended at the start of the whole sequence.

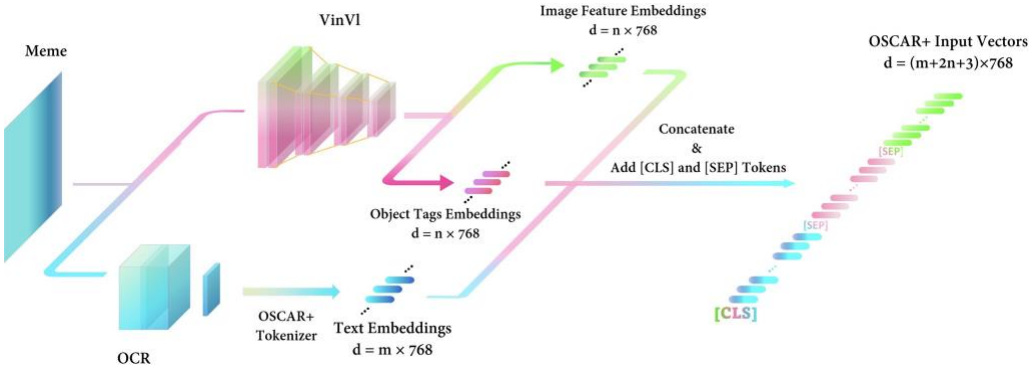

Figure 4: Meme Preprocessing for OSCAR+ Input

1.4.2 Visual-Language PTM

Like many other PTMs, OSCAR+ was based on multi-layer Transformers (**Figure 6** and **7**) [42]. Unlike most existing PTMs which simply concatenate image region features and text features as input and resort to the self- attention mechanism (**Figure 8**) to learn semantic alignments between image regions and text in a brute force manner, OSCAR+ additionally takes into account explicit representation of object tags, serving as the anchor points and grounding for both image features and text features. This is motivated by the observation that the salient objects in an image can be accurately detected by modern object detectors[43], and that these objects are often mentioned in the paired text. This can help to overcome the problem of visual regions being often over-sampled[44], noisy and ambiguous. In this study, we show that the learning of cross-modal representations can be significantly improved by introducing object tags detected in images as anchor points to ease the learning of semantic alignments between images and texts. We propose a new VLP method Oscar, where we define the training samples as triples, each consisting of a word sequence, a set of object tags, and a set of image region features.

The whole OSCAR+ model consists of 12 encoder blocks, and each encoder blocks takes as input the output embedding sequence produced by the previous encoder. The first encoder takes as input the embedding matrix of dimension $(m + 2n + 3) \times 768$ which is produced by stacking the image features embeddings, the object tag embeddings, the text embeddings and the embedding of special tokens (**[cls]**,

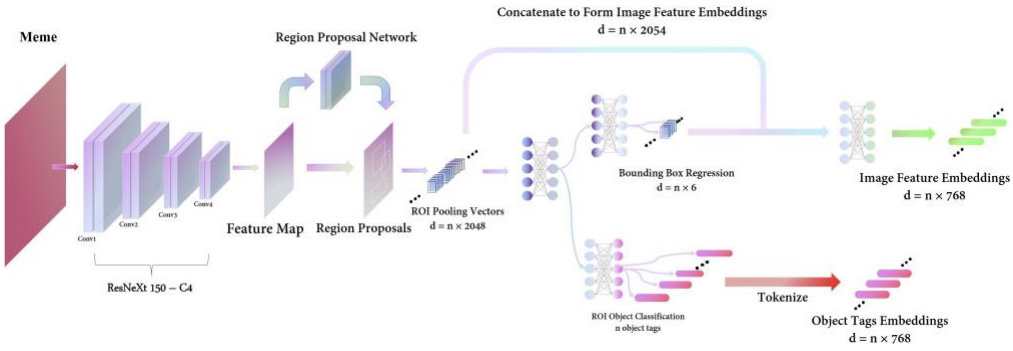

Figure 5: VinVl Architecture

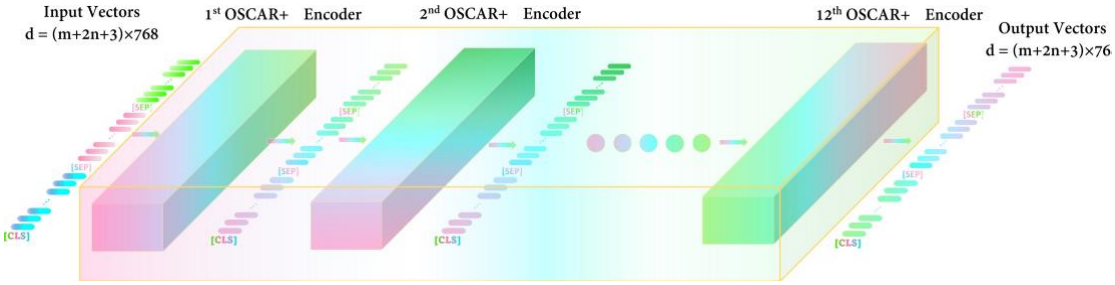

Figure 6: OSCAR+

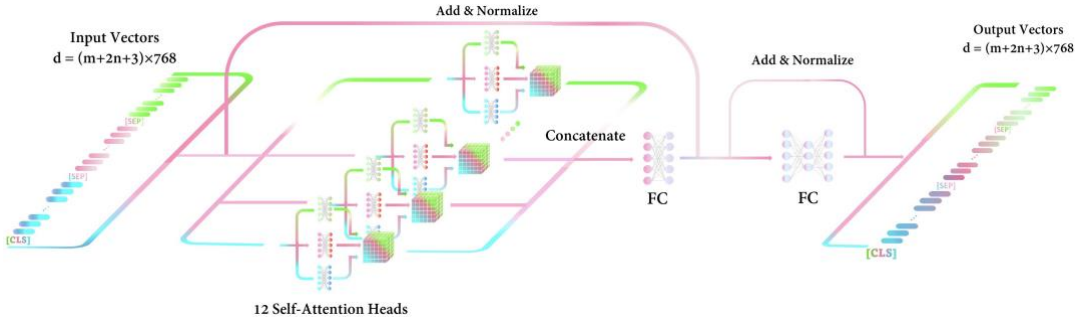

Figure 7: OSCAR+ Encoder

[SEP]) from the pre-processing stage. In every encoder block, the input embedding sequence is passed through 12 self-attention heads (Figure 8) in parallel, each outputting a smaller matrix of dimension $(m + 2n + 3) \times 64$. More specifically, in each self-attention head, the input matrix will pass through three separate FCNN to produce three smaller matrices **Q**, **k** and **V** of dimension $(m + 2n + 3) \times 64$. Then, the standard dot product attention operation was carried out as

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V, \quad (1)$$

where d_k equals to 64. These intermediate output matrices are again concatenated to form a matrix of the original size, passed through an FCNN, then added with the original matrix, and normalized by rows. This normalized matrix is again passed through an FCNN, added with itself, and normalized by rows again.Finally, the encoder block will produce a matrix of dimension $(m + 2n + 3) \times 768$.

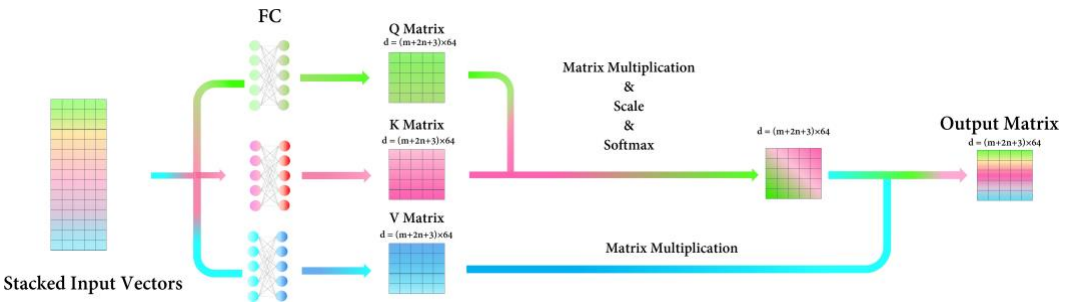

Figure 8: Self-Attention Head

1.5 Results

Model	Validation Acc.	Validation AUROC	Test Acc.	Test AUROC
Image-Grid	50.67	52.33	52.73	53.71
Image-Region	52.53	57.24	52.36	57.74
Text BERT	58.27	65.05	62.80	69.00
MMBT-Grid	59.59	66.73	62.83	69.49
MMBT-Region	64.75	72.62	67.66	73.82
VilBERT	64.75	72.62	67.66	73.82
Visual BERT	65.01	74.14	66.67	74.42
ViLBERT CC	61.40	70.07	61.10	70.0
Visual BERT COCO	65.93	74.14	69.47	75.44
OSCAR+ LC	66.32	75.37	68.52	75.56
OSCAR+ RF	**66.58**	**76.83**	**68.73**	**77.14**

Table 1: Comparisons with different baseline models.

The baseline provided by Kiela et al.[37] including both unimodal PTMs and multimodal PTMs. The unimodal PTMs are BERT [14] (Text BERT), standard ResNet-152 [30] convolutional features from res-5c with average pooling (Image-Grid), and features from fc6 layer that are fine-tuned using weights of the fc7 layer (Image-Region). The multimodal baseline methods include supervised multimodal bi-transformers [45] using either Image-Grid or Image-Region features (MMBT-Grid and MMBT-Region), versions of VilBERT[31] and Visual BERT[46] that were only unimodally pretrained and not pretrained on multimodal data (ViLBERT and Visual BERT)¿ The multimodal baselines are ViLBERT trained on Conceptual Captions[47] (ViLBERT CC) and Visual BERT trained on COCO dataset[48] (Visual BERT COCO). and Visual BERT (trained on COCO, Visual BERT COCO). The results are shown in Table 1. We observe that the text-only classifier performs slightly better than the vision-only classifier, and multimodal PTMs performed significantly better than the unimodal models. Our proposed OSCAR+ models also performed better than other multimodal PTMs. Compared to OSCAR+ with simple linear classifier (OSCAR+ LC), OSCAR+ with random forest classifier (OSCAR+ RF) achieved both higher accuracy and AUROC (Area Under the Receiver Operating Characteristic) in validation set (dev set) and test set (**Table 1**).

1.6 Discussion

This study aimed to classify hateful memes using a specific multimodal PTM and to compare its performance with other simple. The results demonstrate that multimodal PTMs are better than unimodal models, and that by also intaking object tags, PTM can achieve greater accuracy and AUROC. This is because the hateful memes generally involves both visual and textual cues that could only be identified when considering them simultaneously, and because explicit representation of object tags further provided the model with clues of the features that to which it needs to pay more attention. For OSCAR+ specifically, the random forest classifier had better performance than linear classifier because it had greater degree of freedom to separate two groups of data points in high-dimensional representation space while linear classifier simply divide the representation space into two sections. Some of the models' classification false results were presented in **Figure 9** and **10**. We found that the memes that invoked simultaneously the visual and textual cues that complemented each other were difficult to be classified correctly. And we also found that the memes that involves objects with specific external knowledge, such as the symbols of ethnic groups, nations and religions, can also pose a challenge for accurate detection.

Moreover, aside from the aforementioned limitations, the detection models also face many more problems in real-life online public space. For example, modern online communication heavily employs non-standard features such as emojis and other irregular tokens such as $; and hateful users often try toevade detection by substituting the characters in their messages with very symbols which are very different in terms of machine encodings yet look or sound similar to human beings. One future improvementfor our hateful message detection system is to take advantage of these underutilized visual or audio as- pects of the textual information in order to include more real-life scenarios. At the same time, as we showed that detecting underlying hateful metaphors requires the system to possess the ability to relate visual and linguistic entities in the image or captions to the real-world knowledge base, we expect future Visual-Language PTMs that are supplemented with external knowledge base, such as DBpedia[49] and wikidata[50] to achieve better performance on this task. Most of the contemporary pre-trained visual language models only take image and text into account, and the training dataset containing only common

Figure 9: False positive example

Figure 10: False negative example

objects. The hateful memes often invokes objects that are both uncommon and specific, connected to historical or social events that are not present in those training sets. Thus, a sufficient dataset related to social or historical events is also in great demand.

1.7 Conclusion

This study has demonstrated that Visual-Language PTMs are capable of detecting hateful memes that involve strongly correlated textual and visual information. Compared to previous smaller or unimodal detectors, our proposed model has achieved significant improvement in detection performance. We further showed that compared to other multimodal PTMs, OSCAR+ was better performing because it used object tags as anchor points to better corroborate visual and textual information. In addition, we pointed out that our model is still not good enough to detect hateful memes that involves external knowledge, and provided future researchers with a room for further investigation.

References

[1] J. Greenberg and T. Pyszczynski, "The effect of an overheard ethnic slur on evaluations of the target: How to spread a social disease," *Journal of Experimental Social Psychology*, vol. 21, no. 1, pp. 61–72, 1985.

[2] B. Mullen and D. R. Rice, "Ethnophaulisms and exclusion: The behavioral consequences of cognitive representation of ethnic immigrant groups," *Personality and Social Psychology Bulletin*, vol. 29, no. 8, pp. 1056–1067, 2003.

[3] B. Mullen and J. M. Smyth, "Immigrant suicide rates as a function of ethnophaulisms: Hate speech predicts death," *Psychosomatic Medicine*, vol. 66, no. 3, pp. 343–348, 2004.

[4] M. L. Williams, P. Burnap, A. Javed, H. Liu, and S. Ozalp, "Hate in the machine: Anti-black and anti-muslim social media posts as predictors of offline racially and religiously aggravated crime," *The British Journal of Criminology*, vol. 60, no. 1, pp. 93–117, 2020.

[5] F. Fasoli, A. Maass, and A. Carnaghi, "Labelling and discrimination: Do homophobic epithets undermine fair distribution of resources?," *British Journal of Social Psychology*, vol. 54, no. 2, pp. 383–393, 2015.

[6] W. Soral, M. Bilewicz, and M. Winiewski, "Exposure to hate speech increases prejudice through desensitization," *Aggressive behavior*, vol. 44, no. 2, pp. 136–146, 2018.

[7] "Fact check - hate speech." `https://abalegalfactcheck.com/articles/hate-speech.html`. Published: 2017-8-17.

[8] "Facebook Community Standards." `https://transparency.fb.com/policies/community-standards/`. Accessed: 2021-09-28.

[9] "YouTube Policies." `https://transparency.fb.com/policies/community-standards/`. Accessed: 2021-09-28.

[10] "Twitter Hate Speech Policies." `https://help.twitter.com/en/rules-and-policies/hateful-conduct-policy`. Accessed: 2021-09-28.

[11] "THE TRAUMA FLOOR - The secret lives of Facebook moderators in America." `https://www.theverge.com/2019/2/25/18229714/cognizant-facebook-content-moderator-interviews-trauma-working-conditions-arizona`. Published: 2019-2-25.

[12] D. Yin, Z. Xue, L. Hong, B. D. Davison, A. Kontostathis, and L. Edwards, "Detection of harassment on web 2.0," *Proceedings of the Content Analysis in the WEB*, vol. 2, pp. 1–7, 2009.

[13] N. D. Gitari, Z. Zuping, H. Damien, and J. Long, "A lexicon-based approach for hate speech detection," *International Journal of Multimedia and Ubiquitous Engineering*, vol. 10, no. 4, pp. 215–230, 2015.

[14] A. Rodriguez, C. Argueta, and Y.-L. Chen, "Automatic detection of hate speech on facebook using sentiment and emotion analysis," in *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pp. 169–174, IEEE, 2019.

[15] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, "Abusive language detection in online user content," in *Proceedings of the 25th international conference on world wide web*, pp. 145–153, 2016.

[16] Y. Mehdad and J. Tetreault, "Do characters abuse more than words?," in *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pp. 299–303, 2016.

[17] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, "Deep learning for hate speech detection in tweets," in *Proceedings of the 26th international conference on World Wide Web companion*, pp. 759–760, 2017.

[18] Z. Zhang, D. Robinson, and J. Tepper, "Detecting hate speech on twitter using a convolution-gru based deep neural network," in *European semantic web conference*, pp. 745–760, Springer, 2018.

[19] C. M. Lee, "Jigsaw multilingual toxic comment classification use tpus to identify toxicity comments across multiple languages https://www.kaggle.com/c/jigsaw-multilingual-toxic-comment-classification/discussion/160862," 2020. Last accessed September 12 2021.

[20] R. Gomez, J. Gibert, L. Gomez, and D. Karatzas, "Exploring hate speech detection in multimodal publications," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1470–1478, 2020.

[21] J. Devlin, S. Gupta, R. Girshick, M. Mitchell, and C. L. Zitnick, "Exploring nearest neighbor approaches for image captioning," *arXiv preprint arXiv:1505.04467*, 2015.

[22] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, "Vqa: Visual question answering," in *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

[23] B. Zhou, Y. Tian, S. Sukhbaatar, A. Szlam, and R. Fergus, "Simple baseline for visual question answering," *arXiv preprint arXiv:1512.02167*, 2015.

[24] L. Specia, S. Frank, K. Sima'An, and D. Elliott, "A shared task on multimodal machine translation and crosslingual image description," in *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pp. 543–553, 2016.

[25] A. Mogadala, M. Kalimuthu, and D. Klakow, "Trends in integration of vision and language research: A survey of tasks, datasets, and methods," *Journal of Artificial Intelligence Research*, 2021.

[26] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[27] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[28] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *arXiv preprint arXiv:1910.10683*, 2019.

[29] P. He, X. Liu, J. Gao, and W. Chen, "Deberta: Decoding-enhanced bert with disentangled attention," *arXiv preprint arXiv:2006.03654*, 2020.

[30] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding," *arXiv preprint arXiv:1804.07461*, 2018.

[31] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," *arXiv preprint arXiv:1908.02265*, 2019.

[32] W. Su, X. Zhu, Y. Cao, B. Li, L. Lu, F. Wei, and J. Dai, "Vl-bert: Pre-training of generic visual-linguistic representations," *arXiv preprint arXiv:1908.08530*, 2019.

[33] R. Zellers, Y. Bisk, A. Farhadi, and Y. Choi, "From recognition to cognition: Visual commonsense reasoning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6720–6731, 2019.

[34] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh, "Making the v in vqa matter: Elevating the role of image understanding in visual question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6904–6913, 2017.

[35] M. Gardner, Y. Artzi, V. Basmova, J. Berant, B. Bogin, S. Chen, P. Dasigi, D. Dua, Y. Elazar, A. Gottumukkala, *et al.*, "Evaluating models' local decision boundaries via contrast sets," *arXiv preprint arXiv:2004.02709*, 2020.

[36] D. Kaushik, E. Hovy, and Z. C. Lipton, "Learning the difference that makes a difference with counterfactually-augmented data," *arXiv preprint arXiv:1909.12434*, 2019.

[37] D. Kiela, H. Firooz, A. Mohan, V. Goswami, A. Singh, P. Ringshia, and D. Testuggine, "The hateful memes challenge: Detecting hate speech in multimodal memes," *arXiv preprint arXiv:2005.04790*, 2020.

[38] P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang, L. Wang, Y. Choi, and J. Gao, "Vinvl: Revisiting visual representations in vision-language models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5579–5588, 2021.

[39] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, *et al.*, "Oscar: Object-semantics aligned pre-training for vision-language tasks," in *European Conference on Computer Vision*, pp. 121–137, Springer, 2020.

[40] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, pp. 91–99, 2015.

[41] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500, 2017.

[42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[43] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *arXiv preprint arXiv:2104.11892*, 2021.

[44] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6077–6086, 2018.

[45] D. Kiela, S. Bhooshan, H. Firooz, E. Perez, and D. Testuggine, "Supervised multimodal bitransformers for classifying images and text," *arXiv preprint arXiv:1909.02950*, 2019.

[46] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang, "Visualbert: A simple and performant baseline for vision and language," *arXiv preprint arXiv:1908.03557*, 2019.

[47] P. Sharma, N. Ding, S. Goodman, and R. 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)*, pp. 2556–2565, 2018.

[48] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*, pp. 740–755, Springer, 2014.

[49] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "Dbpedia: A nucleus for a web of open data," in *The semantic web*, pp. 722–735, Springer, 2007.

[50] D. Vrandečić and M. Krötzsch, "Wikidata: a free collaborative knowledgebase," *Communications of the ACM*, vol. 57, no. 10, pp. 78–85, 2014.