Supplementary Material document for Explaining models relating objects and privacy

Alessio Xompero[1], Myriam Bontonou[2], Jean-Michel Arbona[2], Emmanouil Benetos[1], Andrea Cavallaro[1,3,4]
[1]Queen Mary University of London, United Kingdom, [2]ENS de Lyon and CNRS, France,
[3]Idiap Research Institute, [4]École polytechnique fédérale de Lausanne, Switzerland
{a.xompero,emmanouil.benetos}@qmul.ac.uk, {myriam.bontonou,jeanmichel.arbona}@ens-lyon.fr,
a.cavallaro@idiap.ch

We provide details of the methods considered for our evaluation and explainability analysis. Specifically, we provide the rationale behind the chosen methods, a review of the main aspects of each method. We also provide details of the parameter settings, and training details, and implementation details in common to all privacy models for a fair comparison under the settings designed in the main paper.

A. Privacy models

A.1. MLP

Tonge et al.'s method [10] uses a convolutional neural network, pre-trained on ImageNet [2], for multi-label object recognition. The feature vector with the confidence of the 1,000 objects is converted into binary values by assigning 1 to the top-k most confident classes and 0 to the all the other classes. The binarised feature vector is used as input to a classifier trained to predict the privacy of an image. Support Vector Machine was used as classifier and k was set to 10 in the study [10,11]. Baranouskaya and Cavallaro [1] defined different input features, such as person presence, person cardinality, outdoor scene, and sensitive features (e.g., violence), and evaluated both a logistic regression and an MLP as privacy models.

Following the ideas and results of these two studies, we devise a baseline that aims to reproduce the method but using the objects and their features as defined in the main paper (see Section 2). Specifically, we replace the multi-label object recognition with the object detector, and the binary feature vector with the cardinality and confidence features. We also use an MLP as privacy classifier given its best-performing results in Baranouskaya and Cavallaro's work [1]. We simply refer to this baseline as MLP.

Alessio Xompero and Myriam Bontonou equally contributed. Myriam Bontonou is also affiliated with Inserm, France, and Jean-Michel Arbona with Univ Lyon and LBMC Lyon, France.

A.2. Graph-based methods

GIP [14] and GPA [9] belong to the two-stage end-to-end training-based category. Both methods model a graph of objects and two additional nodes representing the public and private classes (privacy nodes). The 80 COCO categories [7] are used as objects.

GIP relates objects and privacy nodes with a weighted, undirected, bipartite graph using the frequency of each object with respect to all images labelled as either private or public in a given dataset [14]. A convolutional neural network (VGG-16) is fine-tuned to extracts deep features from the regions of interest localised in an image and associated with the corresponding object node in the graph. The privacy nodes are initialised with the deep features extracted from the whole image by another fine-tuned convolutional neural network (ResNet-101). When objects are not localised in an image, their features are initialised to 0. All features are also complemented with a 1-hot encoding vector to distinguish the privacy nodes, the object nodes, and the object nodes with zero-initialised features.

GPA relates objects with each other by finding at least one co-occurrence of the objects in the dataset, resulting in an unweighted and undirected graph. GPA uses cardinality as object features and initialises the features of the privacy nodes with the logits from a trainable fully connected layer that maps the outputs (logits) of a ResNet-50 pre-trained for scene recognition to the two privacy classes. The scene classifier is also fine-tuned during the training of GPA. Both GPA and GIP use a Graph Reasoning Model [13] to propagate and refine the node features according to the modelled graph structures, and then use a fully connected layer for the final classification. The Graph Reasoning Model consists of three layers of Gated Graph Neural Network [6] and a modified Graph Attention Network [12, 13].

A.3. From end-to-end to a hybrid approach

To adapt the two methods to a two-stage hybrid approach, we decoupled the graph component (Graph Reasoning Model and fully connected layer) from the CNNs, and we initialise the nodes with the cardinality and confidence features obtained from the pre-trained object detector. This means that there is no longer the end-to-end training of the whole pipeline and fine-tuning of the CNNs.

Under our setting, we cannot initialise the privacy nodes of GIP with the high-dimensionality (4,096) feature vectors extracted by ResNet-101 and hence we initialise the features of the two nodes to 0. We refer to this model as GIP$^{\triangle}$ in Table 1 of the main paper. Note that GIP was trained and evaluated only on the Image Privacy dataset [14], whereas we train a new GIP model trained only on PrivacyAlert.

Similarly, we removed the dependency of the scene classifier and the trainable fully connected layer for GPA. Because of the presence of the privacy nodes, we also discard the background category that was included to account for images with no detected objects. We therefore train a model as close as possible to the original implementation[1] where the features of the object nodes are the cardinality and the binary flag[2]. However, we replace the features of the privacy nodes with pseudo-randomly generated values in the interval $[-20, 20]$ according to the range of the logits estimated by the fine-tuned CNN to simulate a non-optimised and non-zero initialisation of the features. We refer to this model as GPA* in Table 1 of the main paper. Note that we also evaluated a variant with zero-initialisation of the features of the privacy nodes and we obtained the same results.

As we noticed a misplacement of the adjacency matrix in the original implementation, we also corrected this error and train a second model. For this second model, we use both cardinality and confidence features, without the binary flag, for a fair comparison with the other models. We refer to this model as GPA$^{\diamond}$ in Table 1 of the main paper. We also tried with either of the two features, as well as using the projection to a higher dimensionality as done for GA-MLP, but all of these models degenerate to predicting a single class.

B. Parameters setting and training details

Object detector. We use YOLOv3 [8], pre-trained on the 80 categories of COCO [7], as object detector. When localising the objects, we allow a maximum of 50 objects for each image while retaining the most confident ones after re-ranking. We also use a minimum threshold of 0.6 and a non-maximum suppression threshold at 0.4. According to the detector settings, we resize images to a resolution of 416×416 pixels. Note that these settings are different from GIP and GPA, which limit the maximum number of regions of interest only to 12. Moreover, GIP used Mask R-CNN [4] as object detector with a threshold of 0.7 on the object confidence and the weighted edges of their modelled graph included images from the testing set (data leakage). On the contrary, GPA used YOLOv3 [8] with a threshold of 0.8 on the object confidence. Our choice to decrease the threshold is to allow the localisation of more objects in an image, increasing the detected categories and the cardinality for more discriminative features. However, the lower threshold can also result in more false positives and affecting the input features of the privacy model that should be designed to handle noisy data.

Training. For reproducibility of models and experiments, we set the seed to an arbitrary value of 789. Note that we do not analyse variations in the performance due to multiple and different seeds, which is beyond the scope of this paper. As training strategy, we follow the recipe of Benchmarking Graph Neural Networks [3]. We use Adam as optimizer [5] with an initial learning rate of 0.001 and without weight decay. We schedule the learning rate to halve if the balanced accuracy of the validation set does not improve for at least 10 epochs (patience). We use early stopping to interrupt the training of the models if the learning rate decreases to a value lower than 0.00001 or the training time lasts longer than 12 hours. In case none of the two conditions is satisfied, we also set the maximum number of epochs to 1,000. Note that we save the model at the epoch with the highest balanced accuracy in the validation split and We use this model for the evaluation on the testing split. Moreover, we set the batch size to 100.

C. Implementation

We implement all models using PyTorch 1.13.1. We use the PyTorch Geometric library for GIP, GPA, and GA-MLP. We trained all models on a Linux-based machine with a NVIDIA GeForce GTX 1080 Ti (12 GB RAM). To ensure the fairness of the benchmark, all methods share the same training and testing software (i.e., only the model is replaced).

Acknowledgements

This work was supported by the CHIST-ERA programme through the project GraphNEx, under UK EPSRC grant EP/V062107/1 and France ANR grant ANR-21-CHR4-0009.

References

[1] D. Baranouskaya and A. Cavallaro. Human-interpretable and deep features for image privacy classification. In *IEEE Int. Conf. Image Process.*, 2023. 1

[1] https://github.com/smartcameras/GPA/

[2] In our experiments, we found that the flag does not provide any contribution to the model.

[2] J. Deng, W. Dong, R. Socher, L.-J. Li, L. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Conf. Comput. Vis. Pattern Recognit.*, June 2009. 1

[3] V. P.. Dwivedi, C. K. Joshi, A. T. Luu, T. Laurent, Y. Bengio, and X. Bresson. Benchmarking Graph Neural Networks. *J. Mach. Learning Res.*, 24(43), 2023. 2

[4] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick. Mask R-CNN. In *Int. Conf. Comput. Vis.*, Venice, Italy, 22–29 Oct. 2017. 2

[5] D. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. In *Proc. Int. Conf. on Learning Represent.*, 2015. 2

[6] Y. Li, D. Tarlow, M. Brockschmidt, and R. S. Zemel. Gated Graph Sequence Neural Networks. In *Proc. Int. Conf. on Learning Represent.*, 2016. 1

[7] 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 *Eur. Conf. Comput. Vis.*, 2018. 1, 2

[8] J. Redmon and A. Farhadi. YOLOv3: An Incremental Improvement. *CoRR*, abs/1804.02767, 2018. 2

[9] D. Stoidis and A. Cavallaro. Content-based Graph Privacy Advisor. In *Proc. IEEE Int. Conf. Multimedia Big Data*, 2022. 1

[10] A. Tonge and C. Caragea. Image privacy prediction using deep features. In *Proc. AAAI Conf. Artificial Intell.*, 2016. 1

[11] A. Tonge, C. Caragea, and A. Squicciarini. Uncovering scene context for predicting privacy of online shared images. In *Proc. AAAI Conf. Artificial Intell.*, 2018. 1

[12] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio. Graph Attention Networks. In *Proc. Int. Conf. on Learning Represent.*, 2018. 1

[13] Z. Wang, T. Chen, J. Ren, W. Yu, H. Cheng, and L. Lin. Deep reasoning with knowledge graph for social relationship understanding. In *Proc. Int. Joint Conf. Artificial Intelligence*, 2018. 1

[14] G. Yang, J. Cao, Z. Chen, J. Guo, and J. Li. Graph-based Neural Networks for Explainable Image Privacy Inference. *Pattern Recognit.*, 105, Sept. 2020. 1, 2