Uncertainty-aware Personal Assistant for Making Personalized Privacy Decisions

GÖNÜL AYCI, Bogazici University, Turkey
MURAT ŞENSOY, Amazon Alexa AI, UK
ARZUCAN ÖZGÜR, Bogazici University, Turkey
PINAR YOLUM, Utrecht University, The Netherlands

Many software systems, such as online social networks enable users to share information about themselves. While the action of sharing is simple, it requires an elaborate thought process on privacy: what to share, with whom to share, and for what purposes. Thinking about these for each piece of content to be shared is tedious. Recent approaches to tackle this problem build personal assistants that can help users by learning what is private over time and recommending privacy labels such as private or public to individual content that a user considers sharing. However, privacy is inherently *ambiguous* and highly *personal*. Existing approaches to recommend privacy decisions do not address these aspects of privacy sufficiently. Ideally, a personal assistant should be able to adjust its recommendation based on a given user, considering that user's privacy understanding. Moreover, the personal assistant should be able to assess when its recommendation would be uncertain and let the user make the decision on her own. Accordingly, this paper proposes a personal assistant that uses evidential deep learning to classify content based on its privacy label. An important characteristic of the personal assistant is that it can model its uncertainty in its decisions explicitly, determine that it does not know the answer, and delegate from making a recommendation when its uncertainty is high. By factoring in user's own understanding of privacy, such as risk factors or own labels, the personal assistant can personalize its recommendations per user. We evaluate our proposed personal assistant using a well-known data set. Our results show that our personal assistant can accurately identify uncertain cases, personalize them to its user's needs, and thus helps users preserve their privacy well.

Additional Key Words and Phrases: Privacy, uncertainty, online social networks

1 INTRODUCTION

Collaborative systems, such as online social networks (OSNs), enable users to share content with others. With the plethora of online content that is being shared, users are faced with the task of *managing* their privacy. Whenever a user is sharing content, she needs to think through whom the content is shared with, whether the content contains elements that would jeopardize her privacy, and so on. Some systems provide settings to configure sharing behavior, e.g., images can be shared only with friends. However, not all images are the same. For example, a user might be comfortable sharing a landscape image publicly, while she might prefer a family image to be shown only to friends. With current systems, identifying whether an image contains certain aspects that could be considered private is left to the user. Moreover, the content may be shared by the user herself and others. For a user to decide whether her privacy is being violated, she needs to check the contents related to her individually.

This work was done before Dr. Şensoy joined to Amazon Alexa AI.
Authors' addresses: Gönül Aycı, Bogazici University, Turkey, gonul.ayci@boun.edu.tr; Murat Şensoy, Amazon Alexa AI, UK, drmuratsensoy@gmail.com; Arzucan Özgür, Bogazici University, Turkey, arzucan.ozgur@boun.edu.tr; Pınar Yolum, Utrecht University, The Netherlands, p.yolum@uu.nl.

1533-5399/2022/1-ART1 $15.00
https://doi.org/10.1145/3561820

This is obviously time-consuming and error-prone. Ideally, a personal assistant (software) could help the user make decisions by signaling whether the content could be private.

Personal assistants help their users make decisions to ease their online interactions. Personal assistants have been used to help users in various tasks, including time management [29], smart homes [6], voice-assistance [12], and so on. Recently, personal assistants have been used for helping users manage their privacy online. Kökciyan and Yolum [20] develop personal assistants to detect privacy violations in OSNs on behalf of their users. They assume that the personal assistant has access to users' privacy preferences through elicitation. Using these preferences and a domain ontology, the personal assistant computes whether others in the OSN share content about the user against her preferences. Kekulluoglu *et al.* [18] and Such and Rovatsos [37] develop techniques to help users reach privacy decisions when a content being shared is owned by multiple users, such as a group image. Both approaches assume that the personal assistants of the users know the privacy preferences of the users and then they apply negotiation techniques to enable the personal assistants to reach a sharing decision that both users are comfortable with.

Because many approaches depend on using users' privacy preferences, there is a tremendous need to learn users' privacy preferences accurately. If a personal privacy assistant can represent the privacy expectations well, then these privacy assistants can help the users in privacy dealings, such as warning the user when the user attempts to share a private content, negotiate with other users on behalf of the user, and so on.

While learning privacy preferences of a user resembles a classical machine learning problem, there are two properties of privacy that make the problem difficult. First, privacy by definition is ambiguous, making it challenging to specify. This makes the pattern that is searched malleable. Second, the users themselves are not always certain about their own privacy preferences and may change their preferences based on other motives [1]. For these reasons, using a traditional predictive model is unreliable as the cost of making a wrong privacy decision is high.

Ideally, the personal privacy assistant should adhere to the followings properties:

- Unobtrusive: The privacy assistant should learn from the sharing behavior of the user without interrupting the user (e.g., asking the user what to share or not) as well as without requiring additional information about the user or the content, such as age or occupation of the user or the tags of a content. Thus, the privacy assistant should only consult the user if necessary.
- Uncertainty-aware: As mentioned above, privacy decisions are many times ambiguous. A personal privacy assistant may not always be able to decide if a content is private or not for the user. The assistant should be uncertainty-aware of this uncertainty and be able to say "I don't know" rather than making an uncertain decision. Hence, it should let the user know that it is uncertain and delegate the decision back to the user.
- Personalized: There are two aspects of personalization that are important for the personal assistant to consider. First, to be able to understand the privacy expectations of its own user. This is important because privacy is subjective and what one user considers private might not be private for another user. Second, each user has a different *risk* associated with making a wrong decision. The risk here refers to classifying a content as private when it should have been public and vice versa. For example, a user might prefer that the personal assistant be risk-averse and classify a content as public when there is even a slight chance that the user would prefer it private.

Existing privacy personal assistants that learn users' privacy preferences do not address the uncertainty of their predictions while making decisions [23, 27, 40] (see Section 2 for details). The idea of considering risk to personalize decisions has been used before but not been coupled with privacy decisions as we have done here [32]. Accordingly, this paper proposes a personal privacy assistant (PURE) that helps its user to make privacy decisions in a personalized way, taking into account the ambiguity of privacy predictions. An important aspect of PURE is that it explicitly calculates the uncertainty of its decisions using evidential deep learning (EDL) [31], which

quantifies the predictive uncertainty of deep neural networks (DNNs). When PURE is uncertain of its decisions, it delegates the prediction back to the user. PURE uses publicly annotated data set to create an initial model for privacy but also factors in *persona* of a user: person's understanding of risk, personally labeled data, and when she should be consulted. In this way, PURE behaves differently for each user to minimize the user's perceived risk of privacy violations. Moreover, PURE does not need to have access to any other private information of the user (e.g., personal details or usage patterns) as well as any of the users in the system, including the relations among users.

The rest of this paper is organized as follows. Section 2 provides a detailed summary of related work in techniques and tools to help users manage privacy. Section 3 explains our approach in detail. Section 4 provides details on the evaluation setup. Section 5 evaluates the proposed approach on a widely used data set and demonstrates the benefits of capturing and exploiting uncertainty. Finally, Section 6 concludes our work with pointers to future directions.

2 RELATED WORK

The literature on approaches that help users manage their privacy is broad. One of the earlier works is due to Fang and LeFevre [7], who introduce a wizard software based on an active learning paradigm. The wizard generates a privacy preference model using extracted features from visible data and communities, and also user input such as asking questions. The wizard recommends privacy preferences to users for different information items on their profiles, such as birthday, address, or telephone number. One of their key findings is that a user's social network structure is an important resource when modeling the user's privacy preferences. This idea has been exploited also by Kepez and Yolum [19], where they propose a machine learning (ML) based model for image privacy prediction. Their framework is based on several attributes about posts such as the sharing time, the location, and the content of the post. They make use of the user's social network to improve prediction. Both of these approaches are important for the privacy prediction task because their approaches use information about the user, her network, and her posts to improve prediction. However, in situations where such external information is not available, there is still a need to make recommendations to the user based only on the content.

Squicciarini *et al.* [35] propose Adaptive Privacy Policy Prediction (A3P) system that predicts a privacy policy for images based on the information available for a given user in the context of social networks. A3P needs a user to specify some privacy policies before making a prediction of privacy policies. When recommending a privacy policy for an image, A3P takes into account significant resources for the privacy concept such as actual image content, metadata, and social circle. A3P consists of two main components: A3P-core and A3P-social. When a user uploads an image, the A3P-core classifies the image first based on their contents and then, updates each category into subcategories based on their metadata (if exists). Then, A3P-core either predicts a policy based on the historical behavior or invokes A3P-social. A3P-social finds a representative privacy policies using user's social circle. While the A3P achieves high accuracy, it makes use of information beyond the images themselves and does not attempt to capture the uncertainty in the prediction as we have done here.

Various approaches have exploited using textual and visual features to train classifiers. An earlier work is by Zerr *et al.* [45], where they identify that combination of textual and visual features produces the best performance in terms of prediction. However, they do not consider personalization or uncertainty. Tran *et al.* [41] propose a privacy framework, called Privacy-CNH that consists of object and convolutional features using a convolutional neural network (CNN) for image privacy detection. Similarly, Squicciarini *et al.* [34] present a learning model to privacy labels of images for binary privacy labels as private and public as well as multi-class privacy labels such as *Only You* or *Family*. They show that combining scale-invariant feature transformation (SIFT) and tag features perform better than the other two or three combinations such as sentiment, RGB, or facial detection. The results of these approaches have been improved by Tonge and Caragea [40], who tackle the same problem

using deep visual semantic and textual features, namely deep tags and user tags. While extracting deep features, they use pre-trained CNN architectures such as AlexNet [22], GoogLeNet [38], VGG-16 [33], and ResNet [13] with Support Vector Machine (SVM) classifiers for the privacy prediction task. Deep tags of images are top k predicted object categories that are extracted from pre-trained models. Using user-created tags, they create deep visual features by adding highly correlated tags to visual features extracted from the fully connected layer of the pre-trained models. Their results show that a combination of user tags and deep visual features from ResNet with the top 350 correlated tags yield the best performance. Moreover, based on their experimental results, fine-tuned networks perform better than learning models trained on the pre-trained features. While their focus has been on classification alone, here, we attempt to take into account both the uncertainty and the personalization associated with making privacy decisions, which is critically important for real-life use cases.

Alternative to approaches that use the image content, some recent approaches have used the tags associated with the content to predict privacy labels of images. Squicciarini *et al.* [36] introduce Tag-To-Protect (T2P) system that automatically recommends privacy policies using the image tags and privacy policies. Their proposed system is useful for both newly uploaded images and cold-start problems when there are very few tags available. One of the prominent results from their experiment is that the prediction accuracy decreases when there is a large set of tags. Since, if the number of tags per image increases, finding a pattern becomes difficult. Kurtan and Yolum [23] propose an agent-based approach that predicts the binary privacy labels of images such as private or public using automatically generated content tags. The system keeps track of content being shared using tag tables. The internal tag table stores the data of privacy labels that are collected from images that the user shares herself. The external tag table stores the data collected from the images that the user's friends have shared with the user. Using metrics inspired by information retrieval, they define metrics to measure how informative a tag is to assess the privacy of an image. Contrary to previous approaches, this system performs well even the personal assistant has access to small data. However, they are not concerned about capturing the uncertainty explicitly or take into account personal risk factors as we have done here.

An alternative set of approaches make use of groups of users, considering various similar aspects among users to make recommendations. Misra and Such [27] develop PACMAN, a personal assistant that recommends access control decisions. Their approach is based on identifying communities (such as friend networks) from the OSN structure of a user and information about the content, such that users manually select tags to extract information about the content. Zhong *et al.* [47] propose Group-Based Personalized Model (GBPM) for an image privacy classification task. Their proposed model learns privacy groups and private content types. Using addition profile information (e.g., gender or age-range), they estimate new users' privacy decisions. They evaluate their proposed model on a randomly selected subset of the PicAlert dataset [45] by first extending it with by adding demographics and social network usage information. They show that GBPM (with profile information) outperforms several baselines such as SVM approaches.

Fogues *et al.* [8] present a personal agent, SoSharP that recommends sharing policies in multiuser scenarios. SoSharP uses contextual-based, user-based, preference-based, and group-based features. These features help to provide personalized recommendations in three rounds. SoSharP makes recommendations to each user by using context-based and user-based features in the first round. It moves to the second round if at least one user has not accepted sharing policy. It uses preference-based features in addition to the features used in the first round. In the final round, it makes a recommendation for all users by using group-based features. As a result of the last round, SoSharP recommends manual resolution if most of the users do not agree with the recommendation. Mosca and Such [28] also propose an agent, ELVIRA, that for multi-user settings that benefits from recommending individual decisions to each user. While we do not consider multi-user settings here, our work can be applied in multi-user settings to recommend privacy labels to each user before a group decision is taken.

Sensoy *et al.* [32] propose risk-calibrated evidential deep classifiers to make a better classification by decreasing the costs of misclassified predictions. They reformulate EDL method in order to accomplish this goal. Their

experiments show that the proposed Risk EDL method has lower misclassification costs compared to EDL, standard learning with cross-entropy loss, and cost-sensitive learning methods for MNIST, FashionMNIST, and CIFAR10 datasets. They also report that their method is robust for out-of-distribution samples. However, they do not apply their model on privacy as we have done here.

Yu *et al.* [43] propose an algorithm to recommend privacy labels of images in OSN. Their recommendation algorithm takes into account two approaches: an image content sensitiveness and trustworthiness of a user. To train a tree classifier, the algorithm uses feature-based and object-based approaches for the image content sensitiveness and characterization of users' trustworthiness based on social behaviors. Through extensive evaluations over user study and two publicly available datasets (such as PicAlert and MIRFLICKR), they have shown that the proposed algorithm is effective. However, PURE is both uncertainty-aware and risk-aware model for predicting privacy labels of images. It can achieve high performance without using the profile information of a user, his/her social connections, or organizing different types of image privacy concerns.

Kokciyan and Yolum [21] propose an approach, TURP, that manages the trustworthiness of information sources, Internet of Things (IoT) devices, for making context-based privacy decisions. They represent IoT devices and users as software agents. Each agent has a confidence value when it shares information with another agent. In the beginning, each device has the same trust value. These values are updated based on feedback that is given from multiple agents. TURP uses Disjunctive Datalog while reasoning about information collected from multiple agents. It would be interesting to couple TURP with our proposed approach in IoT context so that the privacy decisions are augmented with trust.

Jiao *et al.* [15] design a system, IEye, that provides a personalized and interpretable privacy model. They first extract features from images and use multi-layered semantic graphs for feature representations of images. Then, they learn personalized privacy rule sets from images using the rule-based classification algorithm RIPPER. They compare their methods with SIFT and deep features extracted from pre-trained networks AlexNet, VGG16, and ResNet152. They evaluate the performance of their method IEye on the PicAlert dataset and a small dataset called PPP, which consists of 8744 images of 20 users. IEye has a better accuracy result on the PPP dataset than the baseline approaches. However, the proposed method is not better than deep features for the PicAlert dataset.

Yuan *et al.* [44] propose a context-dependent and privacy-aware model for images. The model uses the image's content and contextual information about the image and a specific requester. Their proposed framework first extracts general features (e.g., people, location, time, activity) and contextual features of the sender's images. The system collects information about the sender's preferences by asking questions to the sender in different scenarios. It trains a classifier on this information. When a requester exists, the classifier makes a decision based on the sender's information and the requester's contextual features. They then evaluate their approach to conducting a user study on manually annotated images with personalized contextual sharing decisions through the *ProShare S* application that they developed. However, they do not focus on uncertainty as we have done here.

Dammu *et al.* [3] develop a system for the image privacy prediction task. Their approach is capable of personalization, explainability, configurability, and customizable privacy labels. The system has four modules such as object detection, location detection, object localization, and explicit content extraction. The decision network aggregates outputs of modules for personalized privacy predictions. This comprehensive approach help make personalized prediction of the image labels. To provide such a personalized system, their approach gets feedback from users for misclassified images. Because of the subjectivity privacy, asking users and using their explanations have an important role. However, it is not clear how this would scale in applications that use large image sets.

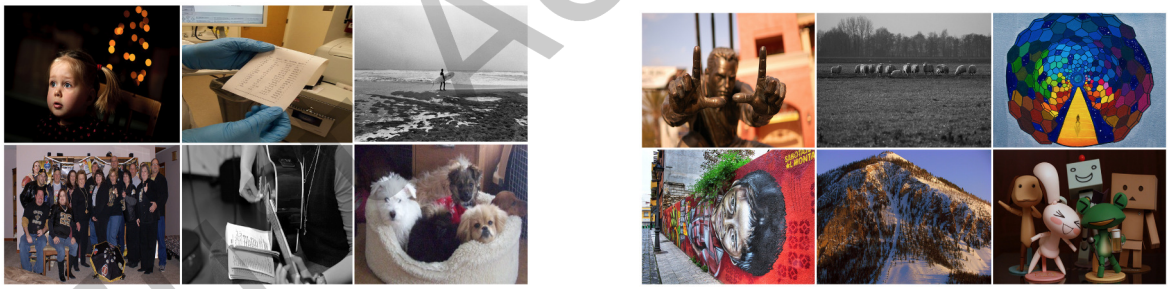
Han *et al.* [11] propose a method which uses multi-level and multi-scale deep representations for the image privacy prediction task. First, they obtain these deep representations CNN based model. Then, they propose two feature aggregation models such as Privacy-MSML and Privacy-MLMS based on different aggregation strategies using Bi-LSTM and self-attention. They evaluate the performance of proposed models on a subset of PicAlert

dataset. They show that their proposed aggregation models yield better performance by F1-score compared with ResNet-18, CNN-RNN, and concatenated multi-level features. However, they are not concerned with capturing the uncertainty in their predictions as we have done here.

Liu *et al.* [25] present problems, challenges, approaches, and future directions of ML in privacy. By taking into account the roles of ML in privacy, they divide existing works into three categories; private ML, ML enhanced privacy protection and ML-based privacy attacks. In the first category, the aim is to develop a private ML system including model parameters, training/test datasets, and predictions. Differential privacy is one of the popular ML solutions that is capable of protecting the privacy of the individual data items [5]. In the second category, ML approaches are used to make decisions for content to preserve privacy and predict information leakage. For instance, ML-based models predict applications' privacy risks, identify contents' sensitive information, and learn users' privacy preferences. The third category presents the importance of ML attack models, including re-identification and inference attacks. Preserving privacy under such attacks becomes more challenging with the rapid increase in the usage of online social networks and the recent advances in DNNs. However, DNNs are vulnerable to adversarial perturbation. Goodfellow *et al.* [10] propose the Fast Gradient Sign method to generate adversarial examples. Differently from such kind of attacks, Miao *et al.* [26] introduce the controlled (protected) information stealing attack. They explain the phases of ML-based attack methodology, discuss the challenges of such attacks, and share the defense mechanisms. The work that we present here falls into the second category, such that we use ML methods to enhance privacy protections.

3 PURE: UNCERTAINTY-AWARE PRIVACY ASSISTANT

We envision PURE to work side-by-side with its user when the user is about to share content and help its user make privacy decisions (Figure 2). PURE uses a learning model that predicts a privacy label of a given image either *private* or *public*. The image is considered to be *private* if it belongs to the private sphere or contains objects that the user cannot share with the world and *public* otherwise. Figure 1 shows examples of images annotated as private and public in the PicAlert dataset by two different annotators.



(a) Private images (b) Public images

Fig. 1. Examples of images labeled as private and public by the annotators.

PURE consists of two modules. The main module is the personalized learning module and serves as the core of the personal assistant. The purpose of this module is three-fold. First, using publicly annotated data, it learns to classify images as private or public. Second, it quantifies uncertainties in predictions, such that when it estimates a prediction to be highly uncertain, it can delegate the decision making to the user. Three, it incorporates the user's expectations in privacy, as each user might have different *persona* when it comes to how they would like to treat certain factors in the learning. The learning model uses evidential deep learning to realize these goals. The learning module produces a classification model that can label a given image and estimate the uncertainty in the

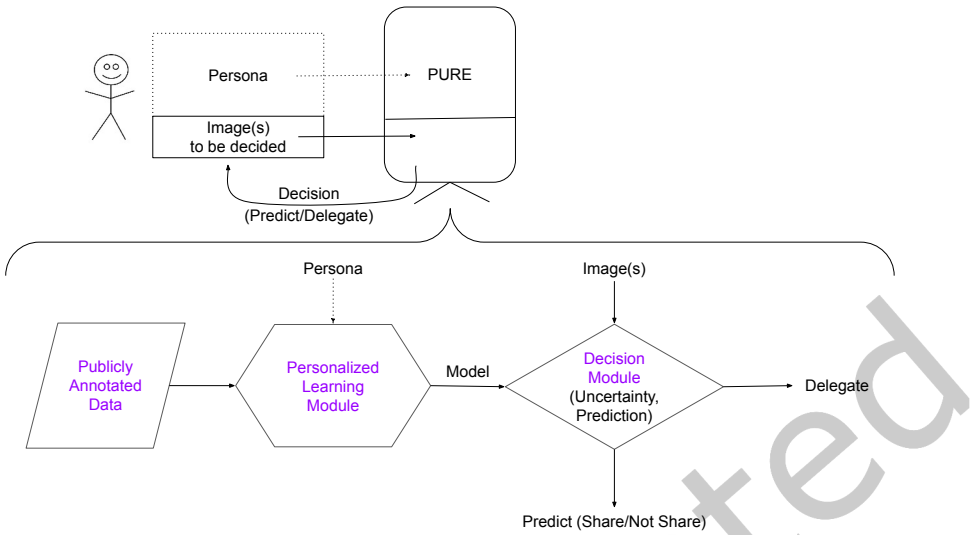


Fig. 2. *System Overview Schema:* OSN user has a personal assistant PURE. She can share persona with her personal assistant. First, PURE has *Publicly Annotated Data* collected from different annotators available in the PicAlert dataset. It learns privacy preferences using visual features in the *Learning Module* and produces *Model*. While learning, the user can share her persona type (i.e., sensitive, semi-sensitive, and non-sensitive) that she can be sensitive about classifying private images as public or not. In this case, the personal assistant is risk-averse. Moreover, the user can share personal data that the user herself annotates allowing learning the user's privacy preferences. Then, PURE makes privacy decisions for its user's content (e.g., image) in the *Decision Module*. While making a prediction for each image, it also generates an uncertainty value for that prediction. To reach a privacy decision, PURE decides whether to use prediction results (i.e., *share* or *not share*) or to delegate the decision to the user (i.e., *delegate*) by comparing the uncertainty value with the threshold received from its user.

prediction. Whenever user provides personally labeled data, this module uses that to tune the personal assistant using the images of the user. This is important because privacy is inherently subjective and the publicly annotated dataset that is used for the learning module may not reflect the privacy expectations of the user. Moreover, due to the subjectivity, PURE might assign high uncertainty to some images. By fine-tuning using personal data, we aim to decrease the uncertainty that PURE might observe with some images. The second module is the decision making module. When a user needs to make a privacy decision, this is the module that is invoked. This module obtains a prediction and an uncertainty value from the model. Each user defines for themselves when to let PURE make a decision and when they would want to be involved. By setting a threshold, a user can choose to decide on the privacy labels when the uncertainty is above the set threshold. Otherwise, the prediction of the model is assigned as the label.

3.1 Learning Privacy Labels with Uncertainty

Evidential Deep Learning (EDL) [31] is based on Dempster-Shafer theory of evidence [4] and subjective logic (SL) [16] to quantify uncertainty in classification tasks. SL expresses degrees of uncertainty through subjective opinions. Each subjective opinion corresponds to a Dirichlet distribution, a conjugate prior for the categorical distribution. For a binary proposition (e.g., the image x is private), the subjective belief of a personal assistant for the truth of this proposition [17, 46] is represented as a binomial opinion, which corresponds to a Beta distribution — a special form of Dirichlet distribution. Since privacy classification is a binary classification task, a personal

assistant's belief for an image to be private is represented as a binomial subjective opinion. A binomial subjective opinion for the classification can be represented as a Beta distribution. That is why, in this section, we will introduce EDL using Beta distributions. Beta probability density function (pdf) is expressed as:

$$Beta\,(p|\alpha,\beta) = \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B\,(\alpha,\beta)} \tag{1}$$

where B is the multivariate beta function [31] and $[\alpha, \beta]$ are the parameters of the *Beta* distribution. In Equation 1, p is the Bernoulli probability that the binary proposition is true, e.g., the probability that the image x is private. A personal assistant has a belief (b) for the proposition that *the image is private*, a disbelief (d) for the same proposition, and an uncertainty (u) that represents the inability to classify the image accurately. The personal assistant's uncertainty about an image may be due to the noise in the image or the lack of training data with similar images. We can calculate these quantities as:

$$b = \frac{\alpha-1}{\alpha+\beta}, \quad d = \frac{\beta-1}{\alpha+\beta}, \quad \text{and} \quad u = \frac{2}{\alpha+\beta},$$

where $b, d, u > 0$ and $b + d + u = 1$. Furthermore, $\alpha - 1$ and $\beta - 1$ are called the evidence for and against the proposition: *the image is private*. Let us note that u is maximized when $\alpha = \beta = 1$, corresponding to the uniform Beta distribution. We can also call them the evidence for the *private* and *public* categories in the classification of the image.

The Beta distribution provides a probability distribution over p — the probability that the given image is private. However, in classification tasks, we need a predictive categorical distribution to decide. For this purpose, we use the expected value of the Beta distribution, which is calculated as follows:

$$\bar{p} = \int_0^1 p\Big(\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B\,(\alpha,\beta)}\Big)dp = \frac{\alpha}{\alpha+\beta} \tag{2}$$

The aforementioned calculations of belief masses and uncertainty are based on the parameters of the corresponding Beta distribution. In order to model belief masses and learn Beta distribution parameters, EDL modifies a vanilla neural network for classification by replacing its softmax layer with a non-negative activation function such as *ReLU*, *softplus*, and *exponential* functions. In our classification problem, we have two categories: *private* and *public*. Given a sample image x, we can use any neural network with two logits outputs: $o_0(x)$ and $o_1(x)$, one for each category. Then, we use the exponential function to calculate evidence for each category as follows: $e_{pub}(x) = exp(o_0(x))$ and $e_{pri}(x) = exp(o_1(x))$, which represent the evidence for the public and private categories, respectively. The Beta distribution parameters α and β for the classification of the image x are calculated as $\alpha(x) = e_{pri}(x) + 1$ and $\beta(x) = e_{pub}(x) + 1$, respectively.

Let $y \in \{0, 1\}$ represent the category index of the sample image x. In standard neural networks for binary classification, the sigmoid function is used to calculate $p(x) = P(y = 1|x)$, i.e., the probability that x is from category $y = 1$. Then, the binary cross-entropy loss is calculated as follows:

$$y \log\Big(p(x)\Big) + (1-y)\log\Big(1-p(x)\Big)$$

There are also other loss functions for classification, such as the Brier score, which is defined as

$$[p(x) - y]^2 + [1 - p(x) - (1-y)]^2. \tag{3}$$

The Brier score is a proper scoring function and is frequently used to measure the accuracy of probabilistic predictions. Unlike vanilla neural classifiers, we do not predict $p(x)$ directly, so we cannot directly use any of these loss functions. However, we predict its Beta distribution $Beta\Big(p(x)|\alpha(x), \beta(x)\Big)$; hence, we may calculate

the expected loss by integrating out $p(x)$ in the classification loss of our choice. We can calculate the expected Brier score for privacy classification as follows:

$$\mathcal{L}(x,y) = \int_0^1 \Big[p(x) - y\Big]^2 + \Big[1 - p(x) - (1-y)\Big]^2 \frac{p(x)^{\alpha(x)-1}(1-p(x))^{\beta(x)-1}}{B\left(\alpha(x),\beta(x)\right)} dp(x) \tag{4}$$

which has the following closed-form solution:

$$\mathcal{L}(x,y) = [\bar{p}(x) - y]^2 + [1 - \bar{p}(x) - (1-y)]^2 + +2\frac{\bar{p}(x)(1-\bar{p}(x))}{\alpha(x)+\beta(x)+1}, \tag{5}$$

where $\bar{p}(x)$ is the expectation of $p(x)$ and calculated as $\alpha(x)/\Big(\alpha(x)+\beta(x)\Big)$ using Equation 2.

$$\mathcal{L}(x,y) = \int_0^1 \Big[y \log\Big(p(x)\Big) + (1-y)\log\Big(1-p(x)\Big)\Big] \frac{p(x)^{\alpha(x)-1}(1-p(x))^{\beta(x)-1}}{B\left(\alpha(x),\beta(x)\right)} dp(x) \tag{6}$$

which has the following closed-form solution:

$$\mathcal{L}(x,y) = y\left(\psi\left(\alpha(x)+\beta(x)\right) - \psi\left(\alpha(x)\right)\right) + (1-y)\left(\psi\left(\alpha(x)+\beta(x)\right) - \psi\left(\beta(x)\right)\right), \tag{7}$$

where $\psi(.)$ is the *digamma* function. We also add a regularizing term $\mathcal{R}(x,y)$ to this loss. $\mathcal{R}(x,y)$ is defined as follows:

$$\mathcal{R}(x,y) = \lambda_t \mathrm{KL}[Beta\Big(p(x);\tilde{\alpha},\tilde{\beta}\Big) \parallel Beta\Big(p(x);1,1\Big)] \tag{8}$$

where

- $t \geq 0$ is the index of the current training epoch,
- $\lambda_t = min\left(1.0, t/10\right)$ is the annealing coefficient,
- KL$[\cdot||\cdot]$ refers to the Kullback-Leibler (KL) divergence,
- $\tilde{\alpha} = \alpha(x)^{1-y} = (e_{pri}(x)+1)^{1-y}$,
- $\tilde{\beta} = \beta(x)^{y} = (e_{pub}(x)+1)^{y}$,
- $Beta\Big(p(x);1,1\Big)$ is the uniform Beta distribution.

Let us note that $\tilde{\alpha}$ and $\tilde{\beta}$ do not contain any evidence supporting the true category of the sample image. That is, $\alpha(x)^{1-y}$ becomes 1 if the image is private ($y=1$) and $\beta(x)^{y}$ becomes 1 when the image is public ($y=0$). As a result, the KL-divergence term is minimized when the network does not produce any evidence for the wrong category. Hence, this regularization term minimizes the evidence generated by the network for the wrong category and increases the predictive uncertainty for the misclassified samples [31].

Example 1. Let's assume that Alice is an OSN user. She has four different images. She needs to decide which images should be shared as public and which should be shared as private. Figure 3 represents an example for predicting privacy labels (such as private or public) of her images and quantifying uncertainty values for each prediction. In Figure 3, the first and the fourth images are public, and the other two are private. As shown in Figure 3, PURE predicts a label for each image as well as an uncertainty value. When producing an answer, it checks its uncertainty value and threshold to decide to answer with its current predicted label or delegate the

Fig. 3. An example for predicting privacy labels of four images and quantifying uncertainty values for each prediction.

decision to its user. If the threshold here is 0.7, it will put forward its predictions for images 2 and 3 (as these have uncertainty values 0.1 and 0.5, respectively) and delegate image 1 and 4 to its user. With this setup, PURE would have correctly classified image 2 but image 3 would have been misclassified. It would have delegated image 1 that it would have failed on, but it would have also been delegated image 4 to the user that this image is correctly classified.

3.2 Personalizing Privacy

Since privacy is inherently subjective, it is important to incorporate personal traits of the user into the decision-making. We consider three aspects of a user that should be factored into the decision making: 1) perception of risk, 2) personal categorization, and 3) preference to be involved.

Perception of risk: While the personal assistant is making decisions, it is possible that it makes a prediction error. It is possible that for some users misclassifying a private image as public may lead to less desirable consequences than misclassifying a public image as private. For some others, there might not be a difference. Furthermore, the cost of different misclassifications may be significantly different for two different users. In order to avoid mistakes that are deemed risky for the user, the system needs to incorporate the risk percention of the user into account.

Typically, vanilla neural networks do not differentiate this significant difference and consider all mistakes as equal. To overcome this, here we introduce a user-dependent risk matrix, which is an asymmetric non-negative square matrix $R \in [0, \infty)^{2\times 2}$. Each value R_{ij} in R represents the user's cost when the classifier assigns an image from category i to the category j. There is no cost for the user for correct classification, hence $R_{ij} \geq R_{ii} = 0$.

There may be different ways of incorporating the user's risk of misclassification into the training of evidential classifiers. In this paper, we propose scaling misleading evidence in the KL-divergence term by modifying $\bar{\alpha}$ and $\bar{\beta}$ as follows:

$$\bar{\alpha} = \left(R_{01} e_{pri}(x) + 1\right)^{1-y}$$
$$\bar{\beta} = \left(R_{10} e_{pub}(x) + 1\right)^{y}$$

This allows us to increase the KL-divergence further when evidence for high-risk categories are produced. The PURE gets R from its user and can learn how to generate evidence for each category based on the personalized

cost of making misclassification. If the user is sensitive about classifying private images as public, the personal assistant also becomes sensitive and avoids generating evidence for the private category for equivocal and ambiguous images.

(1) Scaling misleading evidence in the KL-divergence term by modifying $\bar{\alpha}$ and $\bar{\beta}$ as follows:

$$\bar{\alpha} = \Big(R_{01} e_{pri}(x) + 1\Big)^{1-y}$$

$$\bar{\beta} = \Big(R_{10} e_{pub}(x) + 1\Big)^{y}$$

This allows us to increase the KL-divergence further when evidence for high-risk categories are produced.

(2) Regularizing the amount of misleading evidence directly using the risk of misclassification.

$$(1-y)\bar{p}(x) R_{01} e_{pri} + y\Big(1-\bar{p}(x)\Big) R_{10} e_{pub},$$

where $\bar{p}(x)$ is the predictive categorical distribution calculated as $\bar{p}(x) = \alpha(x)/(\alpha(x) + \beta(x))$. This term will be added to the aforementioned loss: $\mathcal{L}(x, y) + \mathcal{R}(x, y)$.

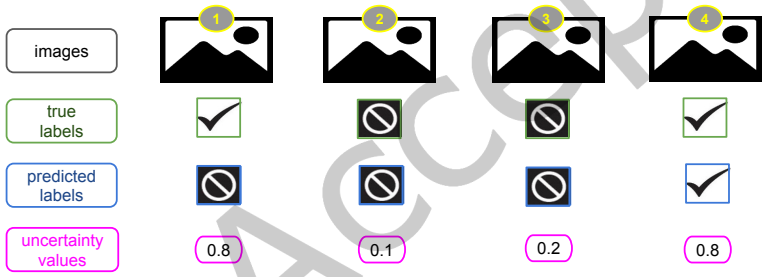


Fig. 4. An example for predicting privacy labels of four images for a sensitive user and uncertainty values for each prediction.

Example 2. If for a user, there is no difference between the misclassification of private and public images, then PURE makes predictions as shown in Figure 3. On the other hand, assume that Alice is more sensitive about classifying a private image as public. By reflecting this in the R_{ij} score, PURE predicts privacy labels of images and quantifies uncertainties for each prediction as shown in Figure 4. Notice that the uncertainty values, as well as the predicted labels, have changed compared to Figure 3. With the uncertainty threshold still set to 0.7, PURE will delegate the same set of images (1 and 4) and answer images 2 and 3. Image 3 has been correctly classified this time. This is a by-product of the fact that PURE chooses to classify more images as private to avoid the potential risk associated with classifying private images as public.

Personal Categorization: Another aspect of personalization is to understand what images a particular user finds private or public. One way of understanding this is to ask the user about privacy preferences. However, there is long standing evidence that users are not good at articulating what they find private. Moreover, their

actions are not always in line with what they claim to be private. Thus, a better way of understanding what is private for a user is to utilize personal data: images that are labeled by the user herself.

PURE makes use of this to fine tune the model it generates. After PURE is trained on publicly annotated data, the user's own labeled data is to adjust the uncertainties in the model. An important contribution of this would be that the uncertainty in certain images drop such that model is more certain of its prediction.

Fig. 5. An example for predicting privacy labels of four images by fine-tuning using personal data and uncertainty values for each prediction.

Example 3. If PURE uses publicly available data and if Alice is a sensitive user about classifying a private image as public, PURE makes predictions in Figure 3 and 4, respectively. Moreover, if Alice shares her personal data that has been annotated by her, PURE will predict privacy labels and uncertainty values for each prediction as shown in Figure 5. If uncertainty threshold is still 0.7, PURE will delegate image 1 to Alice and correctly classify image 2, 3, and 4 (as these have uncertainty values 0.1, 0.2, and 0.3, respectively). Since image 4 would have been correctly predicted with a lower uncertainty value, it would have not delegated to the user. So, all the classifications would be correct this time, and PURE would ask its user less.

Preference to be involved The final part of personalization is to understand how much a user wants to be involved in the decision making. Recent work in HCI show that [2] while some users are happy to have privacy decisions taken by their privacy assistants on their behalf, some users would rather be in the loop. Moreover, this is not always a binary decision in the sense that with some decisions the user might want to be involved while with other she might not. We capture this preference to be involved in the decision making process explicitly using a threshold value θ. Whenever PURE is asked to label an image, in addition to a prediction, PURE also provides a level of uncertainty. When PURE has an uncertainty above θ, it delegates the decision making back to the user. Since θ can be configured by the user herself, it enable the user to select a level of involvement, where $\theta = 1$ would mean letting PURE do all the decisions, where $\theta = 0$ would mean overseeing all the decisions. During our experiments, we discuss having $\theta = 0.7$ as a working setting to capture user involvement only when PURE has high uncertainty.

4 EVALUATION

We evaluate the performance of PURE in terms of its contribution to preserving privacy. Specifically, we aim to answer the following research questions:

RQ1 Does PURE capture the privacy ambiguity through its modeling of uncertainty and by delegating ambiguous cases to the user, can PURE increase its privacy prediction accuracy?
RQ2 Does PURE capture uncertainty adequately and outperform existing models that capture uncertainty?
RQ3 Can PURE enable personalization of privacy by incorporating privacy risks and personal data of the user so that the accuracy is improved for the user while the number of delegated decision decrease?

It is important to be able to answer RQ1 affirmatively because capturing the ambiguity is the key for PURE to choose when to consult its user. Ideally, uncertain images should be delegated to the user for a decision, and certain images should be answered by PURE. As a result, if we consider only the certain images that PURE makes a prediction on, we would expect to obtain a higher accuracy than the overall accuracy. RQ2 investigates the dynamics between uncertainty and making prediction errors and questions whether alternative formulations of uncertainty such as an SNN or well-known uncertainty quantification methods such as MC dropout and Deep Ensemble would suffice. Finally, RQ3 explores if and to what extent personalization of PURE helps users, either in terms of the accuracy they obtain or the number of images they have to decide.

4.1 Dataset

To evaluate our work, we selected a balanced subset of the PicAlert dataset [45]. The PicAlert is a well-known benchmark dataset for the privacy prediction problem for images that contains Flickr images that are labeled as *public* or *private* by external viewers. These images are the most recently uploaded images for four months in 2010 and labeled by 81 users between 10 and 59 years of age with varied backgrounds. 17% of the images in this dataset have conflicting labels from annotators. We consider an image as public if all the annotators have annotated it as public and private if at least one annotator has annotated it as private. The subset we work with contains $32K$ samples that are labeled as *public* and *private*. It is split into *Train* and *Test* sets of $27K$ and $5K$ samples, respectively.

While the previous research aims at increasing the accuracy of the privacy prediction, additionally we focus on how to quantify the uncertainty in these predictions and exploit it to improve the user's privacy in the face of automated decisions.

4.2 Metrics

We evaluate the performance of our approach using two main metrics: (i) success of the model in terms of the standard metrics such as *Accuracy*, *F1-score*, *Precision*, and *Recall*; and (ii) ability of the model to quantify its predictive uncertainty, which allows the improvement of the success metrics in (i) if quantified correctly and accurately.

We first evaluate our approach without considering personalization; hence the generated evidence is not weighted based on the perceived privacy risk of the user. Then, we extend our evaluations with the personalized risk matrices to see how our model adapts itself for users with different misclassification costs. We also extend evaluations with personal data which is annotated by a user to observe how PURE adapts and then asks less to its user. To evaluate the quality of the uncertainty estimates, we calculate the accuracy of the model only on the test samples for which the model's uncertainty is less than a given uncertainty threshold between 0 and 1. When the uncertainty threshold is 1, all test samples are considered in computing the accuracy (and other metrics like precision and recall); however, when the threshold is reduced to 0.5, predictions with uncertainty less than 0.5 are considered for the calculation.

4.3 Evaluation Setting

We use models which are pre-trained on the ImageNet to extract features from images. We compare three popular deep architectures of convolutional neural networks: ResNet50 [13], InceptionV3 [39], and VGG16 [33] in terms of their performance. Table 1 shows the results of the comparison (*Accuracy*, *F1-score*, *Precision*, and *Recall*) of PURE using ResNet50, InceptionV3, and VGG16. ResNet50 and InceptionV3 pre-trained models yield better-performing models as compared to VGG16. ResNet50 avoids the network from the vanishing gradient problem. It has Batch Normalization layers that mitigate Internal Covariate Shift. Because it enables more efficient training and performs well, while having fewer layers and parameters for other models with similar performance, we choose *ResNet50* as our underlying architecture.

	Accuracy	**F1**	**Precision**	**Recall**
ResNet50	0.89	0.89	0.89	0.89
InceptionV3	0.89	0.89	0.89	0.89
VGG16	0.73	0.72	0.8	0.73

Table 1. Performance of PURE using different pre-trained models ResNet50, InceptionV3, and VGG16.

We use the *ResNet50* architecture as our base neural network and replace its last layer (logits layer) with a densely connected layer with two outputs — one for each class (private and public). This architecture has 50 layers with residual connections. We implement our model using Tensorflow and initialize the network layers from the *ResNet50* model and train 15 epochs on the PicAlert dataset using Adam optimizer with a decaying learning rate initialized as $1e-5$. The *ResNet50* accepts images with dimensions $(224 \times 224 \times 3)$, so we resize the images to these dimensions. The implementation used in this work is available at https://git.science.uu.nl/ayci0001/PURE.

5 EXPERIMENTAL RESULTS

We perform the following experiments with the mentioned dataset to answer our research questions.

Performance of PURE: We start with examining the accuracy of PURE, where we configure PURE to provide a label no matter what the uncertainty is. We experiment with using the entire available training data as well as compare it to cases where the training data is smaller.

	Overall			
Usage %	**Accuracy**	**F1**	**Precision**	**Recall**
100	0.89	0.89	0.89	0.89
75	0.88	0.89	0.88	0.88
50	0.88	0.88	0.88	0.88
25	0.87	0.87	0.87	0.87
10	0.79	0.78	0.82	0.79
5	0.66	0.62	0.76	0.66
1	0.55	0.44	0.69	0.55

Table 2. Overall results for PURE as training samples are reduced.

Table 2 shows the overall performance of PURE. For instance, when we use all data while training, PURE obtains an accuracy of 0.89. PURE obtains an accuracy of 0.87 for 25% of training data. On the other hand, if PURE is trained on only 1% of data, the accuracy decreases to 0.55. Table 3 shows the performances of PURE for the private and public classes while using with different amount of data. For instance, PURE achieves F1-score of 0.89 for each class when PURE uses all images in the training dataset. While training with 1%, PURE exhibits poor performance in terms of F1-score, especially for the private class. If a user has only 10% of training data, PURE obtains F1-score of 0.75 and 0.81 for the private and public class, respectively. This is promising because it shows that even when there is limited training data, PURE can be useful.

	Private			Public		
Usage %	**F1**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**
100	0.89	0.91	0.87	0.89	0.87	0.92
75	0.88	0.91	0.87	0.89	0.86	0.92
50	0.88	0.92	0.84	0.89	0.85	0.92
25	0.86	0.91	0.81	0.87	0.82	0.92
10	0.75	0.92	0.63	0.81	0.72	0.95
5	0.49	0.94	0.34	0.74	0.6	0.98
1	0.19	0.85	0.11	0.68	0.52	0.98

Table 3. Results for the private and public classes of PURE at different training sample rates.

Recall that an important aspect of PURE is that it can calculate uncertainty. Next, we look at the relation between uncertainty and accuracy to capture if PURE can represent uncertainty correctly. We have set up PURE so that it would delegate to its user when it is uncertain and thus is likely to make a mistake. Hence, ideally, when PURE delegates to its user, we would expect an improvement in the accuracy of the remaining items.

Table 4 shows the overall performance of PURE with respect to different percentages of delegated predictions. When we do not delegate any predictions, PURE obtains an accuracy of 89% (as was shown in Table 2). When we delegate only 25% of the most uncertain predictions, the results based on all performance metrics improve remarkably, e.g., the accuracy, recall, and precision increase to 0.95. Similarly, when we delegate 75% of the most uncertain predictions, PURE achieves the highest performance of 0.99 in terms of all metrics. Thus, we observe that the delegated images are actually the ones that PURE would have made a mistake in.

	Overall			
Delegation %	**Accuracy**	**F1**	**Precision**	**Recall**
0	0.89	0.89	0.89	0.90
10	0.92	0.92	0.92	0.92
25	0.95	0.95	0.95	0.95
50	0.97	0.97	0.97	0.97
75	0.99	0.99	0.99	0.99

Table 4. Overall results for PURE at prediction delegation rates 0%, 10%, 25%, 50%, and 75% based on uncertainty.

An important question is whether the same upward trend holds for both private and public class. Table 5 shows the performance of PURE for each class. For instance, when PURE does not delegate any predictions, it obtains F1-score of 89% for both private and public classes. When PURE delegates only 25% of the most uncertain predictions, PURE improves F1-scores to 94% and 95% for the private and public classes, respectively. When it delegates 75% of the most uncertain predictions to its user, PURE yields the best performance with 0.99 F1-scores for both private and public classes, respectively. By increasing the number of the delegated predictions, the performance of the model can be improved for each class significantly.

	Private			Public		
Delegation %	**F1**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**
0	0.89	0.91	0.87	0.89	0.87	0.92
10	0.91	0.94	0.89	0.92	0.90	0.94
25	0.94	0.96	0.92	0.95	0.94	0.97
50	0.97	0.99	0.95	0.98	0.96	0.99
75	0.99	0.99	0.98	0.99	0.98	0.99

Table 5. Results for the private and public classes of PURE at various prediction delegation rates (0%, 10%, 25%, 50%, and 75%) based on uncertainty.

Another dimension to understand the link between uncertainty and making errors is to analyze what fraction of wrong predictions fall under different uncertainty rates. Figures 6 and 7 present the uncertainty histogram for the failed and successful privacy prediction of PURE, separately for the private and public classes. The failed and successful predictions in the uncertainty ranges of each class are shown as a percentage among themselves. We observe that failed predictions have higher uncertainty in general while successful predictions are more confident. This indicates that PURE is aware of its own ignorance and possible failures through its predictive uncertainty. When the most uncertain predictions are eliminated, its accuracy improve drastically.

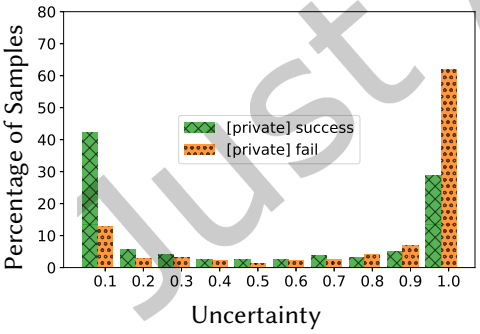


Fig. 6. Uncertainty distribution for the private category.

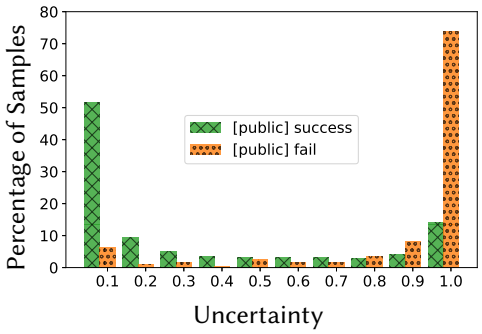


Fig. 7. Uncertainty distribution for the public category.

Similarly, Figure 8 plots uncertainty against accuracy for PURE. The numbers on the particular points denote the ratio of test samples that are decided by PURE; the remaining samples are delegated back to the user because

of the high uncertainty. The case when uncertainty is set to 1 is analogous forcing PURE to make all the privacy decisions, without delegating any case to the user. The accuracy of PURE at this stage is 89%. This is on par with the existing models in the literature that use the same dataset to predict privacy labels [40]. The more interesting cases are the ones where the uncertainty is high so that the PURE decides that there is too much uncertainty to answer and delegates them to the user. For example, for uncertainty threshold 0.4, 57% of the test samples can be decided by PURE, leading to an accuracy around 0.97. For uncertainty threshold 0.8, 69% of the test samples can be decided with PURE, leading to an accuracy around 0.95. This shows, as RQ1 asks, that PURE can capture the privacy ambiguity and it can delegate such cases to its user.

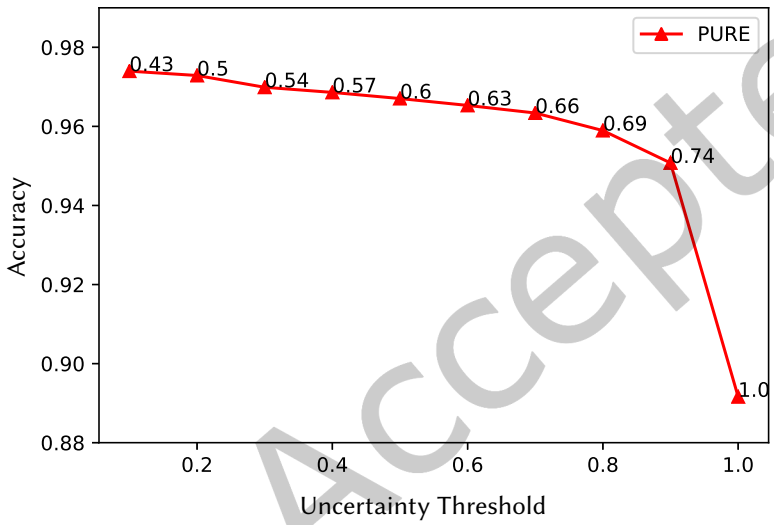


Fig. 8. The change of accuracy with respect to the uncertainty threshold.

Comparison with Alternative Networks: The PURE calculates the uncertainty of its predictions and exploits it to refrain from making wrong privacy decisions for its users. In order to understand the effect of PURE in its calculations of uncertainty, we compare it to alternative predictive uncertainty models; Monte Carlo (MC) dropout [9] and Deep Ensemble [24], which depend on the class probabilities predicted by the neural network as well as a regular Standard Neural Network (SNN). We implement a SNN, MC dropout, and Deep Ensemble with two softmax outputs using the same *ResNet50* architecture with PURE. In order to measure the uncertainty of standard deep classifiers, entropy of their predictions has been used after normalising it to have an uncertainty value between 0 and 1 [14, 31].

To have a meaningful comparison for uncertainty quantification, we use the normalized entropy as a proxy for the uncertainty for the PURE and SNN, MC dropout, Deep Ensemble models.

For PURE, we use the expected probabilities defined in Equation 2 to calculate the entropy. The entropy for the class probabilities p and $(1-p)$ is calculated as $-[p \log p + (1-p) \log(1-p)]$; then normalized by dividing to $\log 2$, which is the maximum entropy for the binary classification. Gal and Ghahramani propose MC dropout method that represents model uncertainty using dropout in neural networks at test time. We add dropout layers after

each non-linearities and set the dropout rate as 0.05 [1]. We train a model, obtain the desired number of different predictions and take the average of 5 predictions for each class. Lakshminarayanan *et al.* propose ensemble based method, called Deep Ensemble that quantifies predictive uncertainty. We use Brier score (Equation 3) as a proper scoring rule as the training criterion, train 5 models with the same architecture, and take the average of predictions.

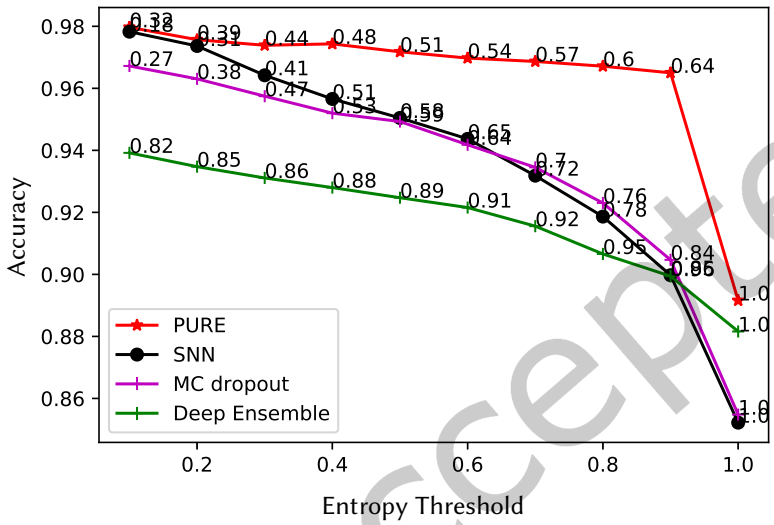


Fig. 9. The change of accuracy for different models with respect to different entropy thresholds.

Figure 9 demonstrates the variation of the test accuracy for the PURE, SNN, MC dropout, and Deep Ensemble models as we change the thresholds for the normalized entropy for delegating predictions. The PURE outperforms the SNN, MC dropout, and Deep Ensemble models at almost all points. Its accuracy is higher than that of alternative models even when there are no delegated predictions and the disparity significantly increases as the predictions are filtered based on their entropy. When we select entropy threshold as 0.4, SNN achieves 95.6% accuracy using 51% of the data. For the same threshold, 53% of the data used by MC dropout, and the accuracy value is 95.2%. Deep Ensemble obtains an accuracy of 93% using 88% of the data, whereas PURE achieves 97.8% of accuracy for 48% of the predictions at the same threshold. Our observation is consistent with the literature, where the deep neural networks are criticized as being overconfident, hence misleading, when they make mistakes [14]. One important aspect to note here is the distribution of the data over various entropy thresholds. In principle, we want to use the entropy threshold to decide if a decision will be delegated to the user. Consider PURE in Figure 9. When entropy is 0.1, PURE will only classify 32% of the data and delegate the remaining to the user. While this is a large percentage to delegate, it comes with the advantage of 98% accuracy. If PURE sets its entropy to 0.8, then it will classify 60% of the data and still yielding an accuracy of 98%. When it chooses to classify all the data, then the accuracy will drop to 89%. Contrast this ability to configure based on entropy to Deep Ensemble. With Deep Ensemble, even when the entropy is set to 0.1, the personal assistant will classify 82% of the data itself, with low

[1] 0.05 yields the best performance among {0.01, 0.1, 0.25, 0.5}.

flexibility in delegating the choices to the user. Next, we study the accuracy changes of these models based on their data usage.

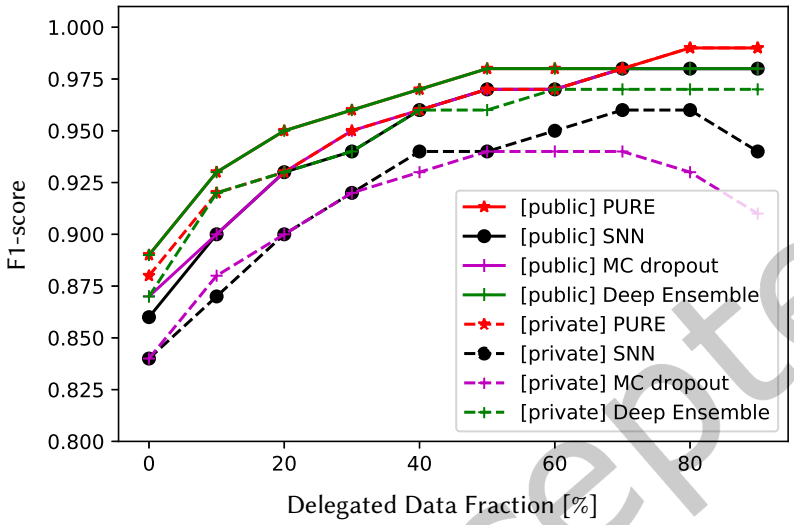


Fig. 10. F1-scores for the private and public classes relative to the percentage of delegated decisions.

Figure 10 plots how the F1-score changes for the *public* and *private* classes when PURE, SNN, MC dropout, and Deep Ensemble models delegate certain percentages of their most uncertain predictions based on the entropy. The F1-scores of PURE and Deep Ensemble models are better than SNN and MC dropout for each class and the gap is bigger for the private class. PURE outperforms Deep Ensemble for both classes when the rate is higher than 0.7. The F1-score of PURE improves further and reaches 0.99 for both private and public classes. However, the F1-score of the private class for SNN and MC dropout decreases when the most uncertain 80% of the data is neglected and the remaining most certain 20% is used for the calculation of the F1-score. The decrease in the F1-score for the private class in Figure 10 indicates that SNN and MC dropout is overconfident while PURE can exploit its well-measured uncertainty to avoid wrong privacy decisions. With randomization tests [30], we can show that the improvements of PURE over existing models is statistically significant (p-value < 0.05). We answer RQ2 positively such that PURE outperforms SNN, MC dropout, and Deep Ensemble by expressing uncertainty better.

Personalized Misclassification Risk: Each user may have a significantly different cost for the misclassification of the private content. In this section, we demonstrate the flexibility of PURE for adapting users' perceived risk of such mistakes and its ability to avoid them by refraining from making privacy decisions when uncertain.

For this purpose, we consider five broad categories of personas: *non-sensitive*, *semi-sensitive* and *sensitive*, which have a different risk matrix R. The non-sensitive user has the same perception of risk for the misclassification of private and public image, i.e., $R_{01} = R_{10} = 1$. This means that, for the non-sensitive user, the KL-term in the loss of the PURE does not weigh the evidence for private and public categories differently. On the other hand, for the semi-sensitive users, misclassifying private image as the public is a few times more unacceptable, i.e., $R_{01} = 1$ and $R_{10} = \{3, 5, 7\}$. We also have the sensitive user that misclassifying private image as the public is ten times more

Persona	Non-Sensitive	Semi-Sensitive			Sensitive
Risk Values	$R_{01} = 1$ $R_{10} = 1$	$R_{01} = 1$ $R_{10} = 3$	$R_{01} = 1$ $R_{10} = 5$	$R_{01} = 1$ $R_{10} = 7$	$R_{01} = 1$ $R_{10} = 10$
[Overall] Accuracy	0.89	0.89	0.90	0.90	0.90
[Overall] Recall	0.89	0.89	0.90	0.90	0.90
[Private] Recall	0.86	0.87	0.89	0.90	0.91
[Public] Recall	0.92	0.91	0.90	0.89	0.89

Table 6. Results for five different risk personas.

	Round I	Round II - Personal	Round II - Random
User 1	0.32	0.19	0.34
User 2	0.30	0.21	0.27
User 3	0.34	0.22	0.29

Table 7. Ratios of prediction whose uncertainty values are greater than 0.7 (θ).

unacceptable for such user, i.e., $R_{01} = 1$ and $R_{10} = 10$. This means that PURE is significantly more penalized for making a wrong prediction in the private class.

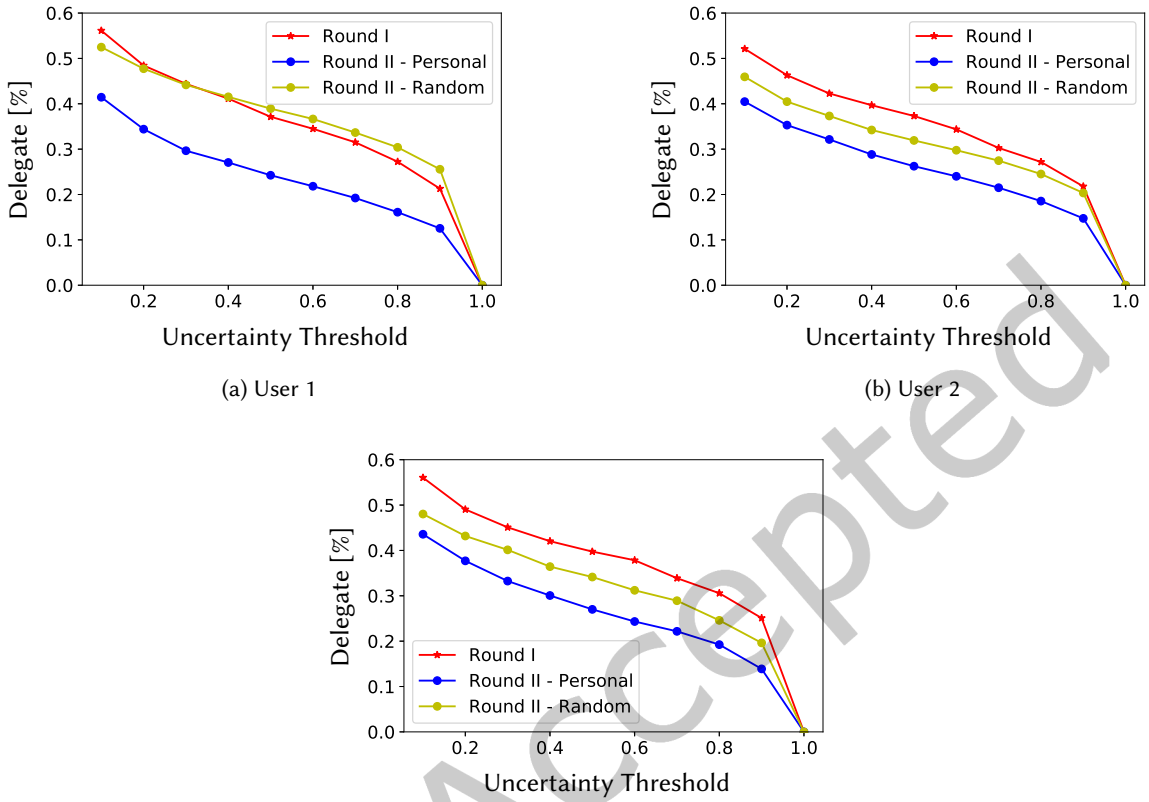
Table 6 shows our results. For the non-sensitive persona, the results are as before. For the sensitive persona, the recall for the private category improves significantly at a cost of having lower recall for the public category. In other words, PURE prefers to classify a content as private over public, when in doubt. This behaviour increases the number of predictions for the private category for the sensitive user (i.e., private recall increase from 0.86 to 0.91). While doing so, it does not sacrifice its overall recall and the accuracy. Our results indicate that by increasing R_{10}, we increase private recall, thus classify more images as private and as a result, we obtain a lower recall for public images. Notice that the R value belongs to a user and can be adjusted as needed.

Data Personalization: PURE delegates a decision to its user when uncertain about a prediction. Ideally, we would like to minimize the number of times this happens while keeping the accuracy high. The personalization is meant to serve this purpose. In order to see if this is indeed achieved, we need to perform a comparative analysis where in the first round no personal data are used and in the second round the personal data are added. To realize this, we select three users who have annotated the most images as each annotator has annotated different number of images.

- **Round I:** Train a model without using personal data.
- **Round II-Personal:** Tune a trained model in *Round I* using personal data annotated by a user.
- **Round II-Random:** Tune a trained model in *Round I* using data annotated by others.

Figure 11 shows the change of samples (%) that PURE prefers to delegate decision of labels to the user. We observe that PURE delegates less to its user when it makes use of the personalization module.

A closer look in Table 7 shows the ratio of samples at each round when uncertainty threshold is 0.7. These samples belong to the top three users who annotate the most images. For instance, for the first user, PURE delegates 32% and 34% of test samples after *Round I* and *Round II - Random*, respectively. However, only 19% uncertain cases can be delegated to the user when we tune the trained model with personal data at Round II. In light of these results, we answer RQ3 positively: PURE can adjust its behavior based on the personal risk and expectations of its user as well as help the user deal with fewer decisions.



(a) User 1

(b) User 2

(c) User 3

Fig. 11. The change of delegated samples for each round with respect to different uncertainty thresholds.

An interesting aspect to note is that the number of images use to personalize can affect the behavior of the data personalization approach. Given the limited number of annotators with predefined set of images, we currently cannot study such questions but it would be useful to provide bounds to guide users to personalize the assistant even further.

6 CONCLUSION

This paper proposes a personal privacy assistant, PURE, that helps its user make privacy decisions by recommending privacy labels (private or public) for given contents. PURE is uncertainty-aware in that it captures the privacy ambiguity using uncertainty modeling and delegates decisions for ambiguous cases to its user. PURE is personalized that it is capable of making a privacy decision by incorporating the user's risk of misclassification and using personally labeled data. Through its personalization, PURE is also unobtrusive as it does not consult its user only when it is uncertain. Our experimental results show that PURE obtains a high accuracy without even consulting its user at all. Our comparison with other models in the literature show that PURE captures uncertainty well and that most of the content that it identifies as uncertain are the ones that it would have made

an error if it were to classify them itself. Thus, the overall accuracy of PURE steadily increases as it delegates uncertain cases to its user. Moreover, our results show that PURE can indeed adjust its behavior based on the personal risk and expectations of its user and is able to decrease the delegations to its user by fine-tuning using personal data.

This paper opens up interesting directions for future research. Currently, we start with an uncertainty-aware model for privacy classification and enhanced it further with users' personalized risk for misclassification. An interesting direction for further research is to enable PURE to have deeper interactions with the user. For example, it could interact with the user to obtain labels for the images that it is uncertain about and further enhance its ability for classification with this new personal data. Similarly, it could attempt to explain the uncertainty to its user so that the user can help in guiding PURE in the right direction. Semantic information about the content, such as tags, could aid in the explanation. Another important direction is to enable interaction between different personal assistants to help create a collaborative environment for preserving privacy. Currently, we assume that the content that a personal assistant decides on belongs to its user alone. However, many times content, such as group images or co-edited documents, might belong to more than one user [42]. Extending PURE to act collaboratively in such settings would be useful. Finally, a user's privacy preferences are many time relational. That is, a user might be fine with sharing a content with a friend but not with a colleague. Our current approach does not capture with whom the content is shared. It would be interesting to learn relation-based sharing behavior of users. Another interesting direction would be to investigate how PURE can learn when to delegate its decisions to the user. Currently, PURE has a preset threshold value θ by capturing user involvement such that it makes a decision when a prediction's uncertainty value is below θ and delegates the decision otherwise. It would be useful if PURE can automatically adjust θ based on user feedback.

7 ACKNOWLEDGMENTS

The first author is supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) and Turkish Directorate of Strategy and Budget under the TAM Project number $2007K12 - 873$. This research was partially funded by the Hybrid Intelligence Center, a 10-year programme funded by the Dutch Ministry of Education, Culture and Science through the Netherlands Organisation for Scientific Research, https://hybrid-intelligence-centre.nl.

REFERENCES

[1] Alessandro Acquisti and Jens Grossklags. 2005. Privacy and rationality in individual decision making. *IEEE security & privacy* 3, 1 (2005), 26–33.

[2] Jessica Colnago, Yuanyuan Feng, Tharangini Palanivel, Sarah Pearman, Megan Ung, Alessandro Acquisti, Lorrie Faith Cranor, and Norman Sadeh. 2020. Informing the Design of a Personalized Privacy Assistant for the Internet of Things. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.

[3] Preetam Prabhu Srikar Dammu, Srinivasa Rao Chalamala, and Ajeet Kumar Singh. 2021. Explainable and Personalized Privacy Prediction. (2021).

[4] Arthur P Dempster. 2008. A generalization of Bayesian inference. *In Classic works of the Dempster-Shafer theory of belief functions* (2008), 73–104.

[5] Cynthia Dwork. 2008. Differential privacy: A survey of results. In *International conference on theory and applications of models of computation*. Springer, 1–19.

[6] Jide S. Edu, Jose M. Such, and Guillermo Suarez-Tangil. 2020. Smart Home Personal Assistants: A Security and Privacy Review. *ACM Comput. Surv.* 53, 6, Article 116 (Dec. 2020), 36 pages.

[7] Lujun Fang and Kristen LeFevre. 2010. Privacy wizards for social networking sites. In *Proceedings of the 19th International Conference on World Wide Web*. ACM, 351–360.

[8] Ricard L Fogues, Pradeep K Murukannaiah, Jose M Such, and Munindar P Singh. 2017. Sosharp: Recommending sharing policies in multiuser privacy scenarios. *IEEE Internet Computing* 21, 6 (2017), 28–36.

[9] Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*. PMLR, 1050–1059.

[10] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).
[11] Yahui Han, Yonggang Huang, Lei Pan, and Yunbo Zheng. 2021. Learning multi-level and multi-scale deep representations for privacy image classification. *Multimedia Tools and Applications* (2021), 1–16.
[12] Johann Hauswald, Michael A Laurenzano, Yunqi Zhang, Cheng Li, Austin Rovinski, Arjun Khurana, Ronald G Dreslinski, Trevor Mudge, Vinicius Petrucci, Lingjia Tang, et al. 2015. Sirius: An open end-to-end voice and vision personal assistant and its implications for future warehouse scale computers. In *Proceedings of the Twentieth International Conference on Architectural Support for Programming Languages and Operating Systems*. 223–238.
[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.
[14] Eyke Hüllermeier and Willem Waegeman. 2019. Aleatoric and epistemic uncertainty in machine learning: A tutorial introduction. *arXiv preprint arXiv:1910.09457* (2019).
[15] Rui Jiao, Lan Zhang, and Anran Li. 2020. Ieye: Personalized image privacy detection. In *2020 6th International Conference on Big Data Computing and Communications (BIGCOM)*. IEEE, 91–95.
[16] Audun Jøsang. 2016. *Subjective logic*. Springer.
[17] Lance Kaplan, Murat Şensoy, and Geeth de Mel. 2014. Trust estimation and fusion of uncertain information by exploiting consistency. In *17th International Conference on Information Fusion (FUSION)*. IEEE, 1–8.
[18] Dilara Kekulluoglu, Nadin Kokciyan, and Pınar Yolum. 2018. Preserving Privacy as Social Responsibility in Online Social Networks. *ACM Transactions on Internet Technology* 18, 4, Article 42 (April 2018), 22 pages.
[19] Berkant Kepez and Pınar Yolum. 2016. Learning privacy rules cooperatively in online social networks. In *Proceedings of the 1st International Workshop on AI for Privacy and Security*. ACM, 3.
[20] Nadin Kökciyan and Pınar Yolum. 2016. PriGuard: A Semantic Approach to Detect Privacy Violations in Online Social Networks. *IEEE Transactions on Knowledge and Data Engineering* 28, 10 (2016), 2724–2737.
[21] Nadin Kökciyan and Pınar Yolum. 2020. TURP: Managing Trust for Regulating Privacy in Internet of Things. *IEEE Internet Computing* 24, 6 (2020), 9–16.
[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems* 25 (2012).
[23] A Can Kurtan and Pınar Yolum. 2021. Assisting humans in privacy management: an agent-based approach. *Autonomous Agents and Multi-Agent Systems* 35, 1 (2021), 1–33.
[24] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems* 30 (2017).
[25] Bo Liu, Ming Ding, Sina Shaham, Wenny Rahayu, Farhad Farokhi, and Zihuai Lin. 2021. When machine learning meets privacy: A survey and outlook. *ACM Computing Surveys (CSUR)* 54, 2 (2021), 1–36.
[26] Yuantian Miao, Chao Chen, Lei Pan, Qing-Long Han, Jun Zhang, and Yang Xiang. 2021. Machine learning–based cyber attacks targeting on controlled information: A survey. *ACM Computing Surveys (CSUR)* 54, 7 (2021), 1–36.
[27] Gaurav Misra and Jose M Such. 2017. Pacman: Personal agent for access control in social media. *IEEE Internet Computing* 21, 6 (2017), 18–26.
[28] Francesca Mosca and Jose Such. 2021. ELVIRA: an Explainable Agent for Value and Utility-driven Multiuser Privacy. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.
[29] Karen Myers, Pauline Berry, Jim Blythe, Ken Conley, Melinda Gervasio, Deborah L McGuinness, David Morley, Avi Pfeffer, Martha Pollack, and Milind Tambe. 2007. An intelligent personal assistant for task and time management. *AI Magazine* 28, 2 (2007), 47–47.
[30] Eric W Noreen. 1989. *Computer-intensive methods for testing hypotheses*. Wiley New York.
[31] Murat Sensoy, Lance Kaplan, and Melih Kandemir. 2018. Evidential deep learning to quantify classification uncertainty. In *Advances in Neural Information Processing Systems*. 3179–3189.
[32] Murat Sensoy, Maryam Saleki, Simon Julier, Reyhan Aydogan, and John Reid. 2021. Misclassification Risk and Uncertainty Quantification in Deep Classifiers. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2484–2492.
[33] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).
[34] Anna Squicciarini, Cornelia Caragea, and Rahul Balakavi. 2017. Toward automated online photo privacy. *ACM Transactions on the Web (TWEB)* 11, 1 (2017), 1–29.
[35] Anna Cinzia Squicciarini, Dan Lin, Smitha Sundareswaran, and Joshua Wede. 2014. Privacy policy inference of user-uploaded images on content sharing sites. *IEEE transactions on knowledge and data engineering* 27, 1 (2014), 193–206.
[36] Anna Cinzia Squicciarini, Andrea Novelli, Dan Lin, Cornelia Caragea, and Haoti Zhong. 2017. From Tag to Protect: A Tag-Driven Policy Recommender System for Image Sharing. In *2017 15th Annual Conference on Privacy, Security and Trust (PST)*. IEEE, 337–33709.

[37] Jose M. Such and Michael Rovatsos. 2016. Privacy Policy Negotiation in Social Media. *ACM Transactions on Autonomous and Adaptive Systems* 11, 1, Article 4 (Feb. 2016), 29 pages.
[38] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1–9.
[39] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2818–2826.
[40] Ashwini Tonge and Cornelia Caragea. 2020. Image privacy prediction using deep neural networks. *ACM Transactions on the Web (TWEB)* 14, 2 (2020), 1–32.
[41] Lam Tran, Deguang Kong, Hongxia Jin, and Ji Liu. 2016. Privacy-cnh: A framework to detect photo privacy with convolutional neural network using hierarchical features. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30.
[42] Onuralp Ulusoy and Pınar Yolum. 2021. PANOLA: A Personal Assistant for Supporting Users in Preserving Privacy. *ACM Transactions on Internet Technology* 22, 1, Article 27 (2021), 32 pages.
[43] Jun Yu, Zhenzhong Kuang, Baopeng Zhang, Wei Zhang, Dan Lin, and Jianping Fan. 2018. Leveraging content sensitiveness and user trustworthiness to recommend fine-grained privacy settings for social image sharing. *IEEE transactions on information forensics and security* 13, 5 (2018), 1317–1332.
[44] Lin Yuan, Joël Theytaz, and Touradj Ebrahimi. 2017. Context-dependent privacy-aware photo sharing based on machine learning. In *IFIP International Conference on ICT Systems Security and Privacy Protection*. Springer, 93–107.
[45] Sergej Zerr, Stefan Siersdorfer, Jonathon Hare, and Elena Demidova. 2012. Privacy-aware image classification and search. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 35–44.
[46] Jie Zhang, Murat Sensoy, and Robin Cohen. 2008. A detailed comparison of probabilistic approaches for coping with unfair ratings in trust and reputation systems. In *2008 Sixth Annual Conference on Privacy, Security and Trust*. IEEE, 189–200.
[47] Haoti Zhong, Anna Cinzia Squicciarini, David J Miller, and Cornelia Caragea. 2017. A Group-Based Personalized Model for Image Privacy Classification and Labeling.. In *IJCAI*, Vol. 17. 3952–3958.