# Online Attacks on Picture Owner Privacy

Bizhan Alipour Pijani, Abdessamad Imine, Michaël Rusinowitch

# Online Attacks on Picture Owner Privacy*

Bizhan Alipour Pijani, Abdessamad Imine, and Michaël Rusinowitch

[1] Lorraine University, Cnrs, Inria, 54506 Vandœuvre-lès-Nancy, France
bizhan.alipourpijani@loria.fr, abdessamad.imine@loria.fr, rusi@loria.fr

**Abstract.** We present an online attribute inference attack by leveraging Facebook picture metadata (i) alt-text generated by Facebook to describe picture contents, and (ii) comments containing words and emojis posted by other Facebook users. Specifically, we study the correlation of the picture's owner with Facebook generated alt-text and comments used by commenters when reacting to the image. We concentrate on gender attribute that is highly relevant for targeted advertising or privacy breaking. We explore how to launch an online gender inference attack on any Facebook user by handling online newly discovered vocabulary using the retrofitting process to enrich a core vocabulary built during offline training. Our experiments show that even when the user hides most public data (e.g., friend list, attribute, page, group), an attacker can detect user gender with *AUC* (area under the *ROC* curve) from *87%* to *92%*, depending on the picture metadata availability. Moreover, we can detect with high accuracy sequences of words leading to gender disclosure, and accordingly, enable users to derive countermeasures and configure their privacy settings safely.

**Keywords:** Social Network, Attribute Privacy, Online Inference Attack.

## 1 Introduction

Facebook is the dominant platform for users to interact, share pictures, and stay connected to their friends. A Facebook user has a profile, a list of friends, and a digital record of behaviors. For instance, a user's behavioral data comprise the list of pages liked by the user. Attribute inference attacks are emerging threats to user privacy in social media such as Facebook. Previous attribute inference attacks focus on inferring a user's private attributes (e.g., location, gender, sexual orientation, and/or political view) by leveraging either social friendship structures [14] or user behaviors [1]. Attribute inference attacks can be significantly improved by natural language processing techniques (*NLP*) since most Facebook data is textual or can be represented as a text. The recent development of deep learning for *NLP* allows one to capture semantic relations between words from their vectorial representations, lead to an efficient attribute inference attack [22]. The approach also applies to non-textual communication modes by using *Emoji2vec* [10].

* This work is supported by DIGITRUST (http://lue.univ-lorraine.fr/fr/article/digitrust/).

The problem of inferring user's attributes becomes qualitatively different if social structures and user behaviors are not available, which is often the case in the real scenario. Unlike previous studies, we show how to detect Facebook user's gender *online* through his/her shared images along with Facebook generated alt-text and received comments underneath those pictures. In online attacks: (i) the attacker relies on offline analysis knowledge to predict new target user attributes, and (ii) input data are collected online by an attacker crawling the targeted profile. The attacker constructs the offline analysis knowledge by collecting profiles with known attributes (in our case gender) and employs sophisticated techniques (e.g., *NLP*) to capture patterns and structures from collected data. Moreover, Facebook users can build their offline analysis knowledge and consider themselves as online targeted users to check their vulnerability to attribute inference attacks.

Machine learning classifiers, and/or vector representation techniques accurately infer private attributes from users' public data (e.g., page likes). Inferred attributes can be employed to deliver personalized advertisements [5] or privacy breaking [4]. In [11], the authors investigate Facebook users' privacy awareness and show one-half of the *479k* examined Facebook users hide their gender. Facebook users prefer to hide their gender for two reasons. First, they want camouflage against sexual harassment and stalking. The Facebook search bar lets users track down pictures of their female friends, but not the male ones [16]. Second, they want to reduce discrimination. The American Civil Liberties Union $(ACLU)$[1] accused Facebook of enabling employers to use targeting technology that excludes women from receiving job ads for some positions.

While many Facebook users hide their sensitive attributes (e.g., gender, age, political view), pictures are still available to public. A social media sharing analysis conducted by *The New York Times* revealed that *68%* of their respondents share images to give people a better sense of *who they are* and *what they care about*. Users in social media share pictures to receive feedback for their activities, especially from friends, and acquaintances, provide a great sense of connectedness. However, they lose privacy control on their posted pictures due to extra information (i.e., meta-data) added by third-party during the publication process. For any uploaded photos, Facebook implemented an object detection system to provide automatically a set of tags, called alt-text, that describe pictures content. They propose this technique to generate a description that can be used by a text reader for blind users. An attacker can use these tags to relax image processing tasks. Furthermore, when observing a picture on Facebook, people write instinctive comments to express their feeling. Automatically generated alt-text and comments (picture metadata) contain potentially sensitive information available to an attacker.

***Problem Statement.*** We propose a method to infer online the target user gender by using non-user generated data. This method even applies to Facebook users who are cautious about their privacy and hide any type of available in-

[1] `https://www.aclu.org/blog/womens-rights/womens-rights-workplace/facebook-settles-civil-rights-cases-making-sweeping`

formation (e.g., friend list, liked pages, groups, and attributes) on their profile. Since our training dataset only contains *25,456* unique words, the input data of an online attack may contain words that do not occur in this training dataset. The new words are called *out of vocabulary words* ($OOV$). To circumvent this problem, we rely on the pre-trained vectors of an advanced *NLP* model, namely *word2vec*[18], and its version dedicated to emojis *emoji2vec* [10]. *Word2vec* and *emoji2vec*, abbreviated by *WE2V*, are trained on large datasets (e.g., Wikipedia for *word2vec*) with specific writing structure or usage. Therefore their pre-trained vectors should be adapted when we aim to apply them to a specific domain such as Facebook. *Retrofitting* technique [12] is called for adjusting the *WE2V* pre-trained vectors by combining external knowledge (*WE2V* dataset), and internal knowledge (offline collected words/emojis co-occurrence). A simple and direct approach to handle an out of vocabulary word would be to replace it by a synonym. However, this approach fails for our gender inference problem, as the word and its synonym can orient to different genders. An example taken from our dataset illustrates this point: male-posted pictures receive more comments containing the word *gorgeous*, while a synonym of this word, namely *beautiful*, is used more frequently for commenting female posted pictures. To that end, we use cosine similarity score [18] to compute similarities among words/emojis, including $OOV$ words. Our approach assumes the following hypotheses:

1. The commenter's gender is hidden. As a consequence, standard homophily-based methods do not apply.
2. The target user is careful enough to conceal gender information in his/her posted comments. Therefore it is needless to process the comments written by the target user.
3. The user profile name does not disclose gender information as Facebook users often use pseudos.

***Contributions.*** Our contributions and improvements over previous works are:

1. A new online attribute inference attack based on non-user generated data composed of alt-text created by Facebook to describe the content of the picture, and comments containing words and emojis posted by friends, friends of friends, or regular users.
2. A strategy for adapting pre-trained vectors for Facebook by exploiting offline scraped comments.
3. A privacy-enhancing system that pinpoints received comments or posted pictures leading to a gender inference attack.

***Outline.*** The paper is organized as follows: we review related work in Section *2*. In Section *3*, we overview the system architecture. Section *4* defines the gender inference attack. Section *5* presents in detail the offline attack steps. Section *6* presents the online attack steps. Section *7* shows experimental results. In Section *8*, we discuss the attack process, and we conclude in Section *9*.

## 2 Related Work

Profiling has gained great attention in the past decade. Deriving user gender, for instance, is important for recommendation systems. Recently, researchers have investigated social media platforms in order to distinguish males and females from content sharing [8] and behavior [17]. Prior works claim that gender prediction is possible from the writing style [13], word usage [25] of the target user. Gender inference from the target user name can be performed across major social networks [15]. However, the performance of this type of attack is biased towards countries of origin [24]. The authors in [7] propose user gender identification through user shared images in Fotolog and Flickr, two image-oriented social networks. They perform image processing on each crawled image (in offline mode), which is not feasible with online attacks. The diversity and global usage of emojis lead researchers to analyse emoji usage according to gender. The authors in [6] collect the data with *Kika Keyboard* and investigate user preferences. This work has two drawbacks that degrade the performance: (i) opposite-gender friends interaction may affect user emoji usage [20], and (ii) user cautiousness in choosing the emojis. Our work is different in two senses. First, we skip the target user emoji usage and rather rely on other Facebook users' emotional responses to solve the above limitations. Second, we exploit the idea that the picture content has a powerful impact on individuals' emotional responses.

To sum up, in contrast with previous works, we study gender inference attacks on Facebook by considering words, and emojis preferences of other Facebook users (e.g., friends) when commenting pictures published by the target user. We do not explore the user network, which has two advantages: (i) makes the attack feasible even when target personal data and his/her ego-network is unavailable, and (ii) makes the attack suitable for online mode. We showed the benefit of non-user generated data analysis to infer the picture owner's gender [2,3]. This work is different from our previous works. First, our attack is not limited to textual language, or emojis as we combine words, and emojis. Second, we propose an online gender inference attack. Third, we leverage sophisticated words, and emojis vector representation, *word2vec*, and *emoji2vec*, to handle out of vocabulary words, and emojis.

## 3 Architecture

Figures 1 and 2 depict the overall architecture of our system. First, we overview the offline training components, and next, we present our online attack ingredients.

***Offline Training.*** This procedure combines domain specific and external knowledge in the following way (see Figure 1). *Data Crawling* collects Facebook users' data in an offline mode for training gender classifiers. Then *Data Preprocessing* prunes, cleanses and normalizes the collected data. *Feature Extraction and Feature Selection* derive a set of features that contribute the most to gender inference from an initial set obtained by n-grams and correlation of alt-text and

comments. *Retrofitting* is the process of adjusting an existing word/emoji vector representation using extra knowledge (in our case, offline collected words/emojis co-occurrence). It allows us to fit *WE2V* word vector representations to our specific domain, namely Facebook. *Machine Learning* aims to select the best gender classifier among the one that we have trained, using standard evaluation metrics. We discuss in detail all the steps in Section 5.

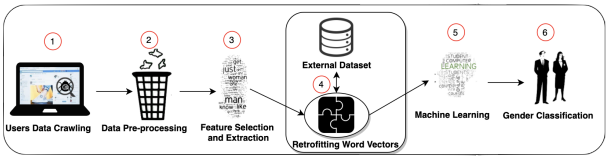



Fig. 1: Offline training

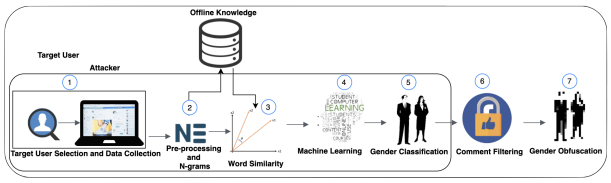



Fig. 2: Online attack and gender obfuscation

***Online Attack and Gender Obfuscation.*** After training the machine learning classifiers the following operations are performed online: (i) gender inference attack by following Steps *1* to *5*, and (ii) user protection from gender inference attack by applying Steps *1* to *7* (see Figure 2). *Target User Selection and Data Collection* selects a specific user and collects his/her data in an online mode. *Pre-processing and N-grams* prunes raw data and extracts features compose of words/emojis or sequence of words/emojis. *Word/Emoji Similarity* finds similarity between word/emoji vectors in the vector space representation. *Gender Classification* applies to the target user based on extracted features. *Comment Filtering* proposes to hide received comments or published pictures that considered to be sensitive to gender. Finally, after hiding the suggested comments, *Gender Obfuscation* re-runs the gender inference attack steps to check whether the gender information secretes. These steps are detailed in Section 6.

## 4 Attack Description

In this work, we consider an attacker who intends to infer a picture owner gender $g$ by observing a set of pictures $P$. Each published picture $p$, where $p \in P$

contains metadata (a set of comments $c_p$, and generated alt-text $a_p$). The attacker can be anyone who can crawl data from Facebook. To accurately infer the target user's gender, the attacker has to cover three possible scenarios. In the first scenario, pictures receive no comments, or comments are unavailable due to privacy reasons (the target user conceals all comments underneath the picture). The attacker can still learn female and male preferences in picture sharing style, and infer target gender from tags or alt-texts. In the second scenario, Facebook is unable to generate automatic alt-texts due to the quality of the posted picture. The attacker learns words/emojis usage from comments posted by other Facebook users to infer the target gender. In the third scenario, both comments and alt-text are publicly available. The attacker can leverage both Facebook users' words/emojis usage and generated alt-text (tags) to infer the target gender. The gender inference attack is based on computing *Mutual Information* ($MI$) on all users labeled by gender $g$ given Facebook generated alt-text $a$, and/or other Facebook users posted comments $c$ for each picture $p$. Let $X$ be a random variable that takes value *1* when the posted photo contains $a$ in the first scenario (resp., $c$ in the second scenario, resp., both $a$ and $c$ in the third scenario) and *0* when the published picture does not contain $a$ in the first scenario (resp., $c$ in the second scenario, resp., both $a$ and $c$ in the third scenario). Let $Y$ be a random variable that takes value *1* (resp., *0*) if the picture owner is female (resp., male). Then, we compute MI as follows:

$$MI(X;Y)_{ac} = \sum_{x\in\{0,1\}} \sum_{y\in\{0,1\}} P(X=x, Y=y)_{ac} log_2 \frac{P(X=x, Y=y)_{ac}}{P(X=x)_{ac} P(Y=y)_{ac}} \quad (1)$$

where $P(X=x)_{ac}$ and $P(Y=y)_{ac}$ are the marginal probabilities of $x$ and $y$, and $P(X=x, Y=y)_{ac}$ is their joint probability. Based on $MI$, the most likely target gender for a set of pictures $P$ is:

$$\arg\max_{g\in\{0,1\}} MI(X;Y)_{ac} \quad (2)$$

which is the core concept behind our inference attack. Let $x_u^i$ be the number of occurrence of feature $i$ in user pictures metadata (a or c). $MI$ measures the mutual dependence between picture owner gender and generated alt-text (resp., received comments, resp., alt-text and received comments) in the first scenario, (resp., second scenario, resp., third scenario). Each user $u$ with set of pictures $P$, and gender $g$ can be represented by a feature set $x_u = \{x_u^1, x_u^2, .., x_u^n\}$, and the label (or class) $y_u$, where $y_u \in \{0, 1\}$. In the offline mode, the attacker trains machine learning algorithms with samples $(x_u, y_u)$, for all $u \in U_{training}$ as inputs, where $x_u = \{x_u^1, x_u^2, .., x_u^n\}$ and $U_{training}$ is a set of users. In the online mode, the attacker carries out the attack on a chosen target user $u_{new}$, by using the features obtained from trained algorithms. We discuss feature selection and extraction techniques in Subsection 5.3.

## 5 Offline Training

In this section, we introduce and discuss the offline components that we implement to train our machine learning algorithms.

### 5.1 Data Crawling from User Context

Let $U$ be the set of Facebook users, where $u_i$ is the $ith$ user with a set of posted pictures $P_i = \{p_i^1, p_i^2, .., p_i^n\}$. For user $u_i$, we extract $\langle g_i, A_i, C_i \rangle$ where $g_i$ is the user gender, $A_i$ is a set of alt-text presented by Facebook and $C_i$ is the set of comments posted by other Facebook users for $P_i$.

### 5.2 Data Pre-processing

Difficulty in analyzing data from social media raises from the presence of different kinds of textual errors, such as misspellings and grammatical errors. In addition to pre-processing steps detailed in [2,3], we perform spelling correction to correct word's spelling and/or intentionally repeated characters (as in *soooooooo beautiful*). However, not all misspelled words can be handled by *NLP* spell correction techniques. For example, *love u* contains a deformation letter *u* that refers to pronoun *you* that can be considered as a misspelled letter *a*, or *luv* is an abbreviation form of *love you*. We re-formulate misspelled words that cannot be corrected by spell correction techniques.

### 5.3 Features Selection and Extraction

Feature selection is the process of identifying and selecting relevant features correlated to variables of interest (gender, in our case). The purpose of feature selection is three-fold: promoting the model prediction performance, providing faster and efficient classifiers, and reducing the data dimensionality to decrease the model complexity. We select features in two different ways:

***1. Contiguous Sequence of Word/Emoji.*** We compute n-grams to capture the occurrences of words/emojis in comments, and tags in alt-text in a given window size ($n$) for each gender. Our experiments revealed that *4-grams*, and *5-grams* are best suited for comments, and alt-text, respectively [2]. By retaining only $n$-grams that appear more than *50* times in total, we collect *2797* features. Table 1(a), and (b) show the discriminative alt-texts generated for female and male-owned pictures, and Facebook users' words/emojis preferences while commenting them, respectively. Additionally, we compute the probability of a person being male or female, given the picture generated alt-text, or words/emojis. For example, $p(female|$😍$)$ is the probability that the user is female if she receives comments with 😍 emoji from other Facebook users.

***2. Correlation of Alt-text and Comments.*** We construct a co-occurrence matrix to find the correlation between gender, alt-text, and received comments

(a)

| *alt_text* | *MI* | *p(female—alt_text )* | *p(male—alt_text)* |
|---|---|---|---|
| closeup | 0.020 | **0.79** | 0.21 |
| smiling | 0.012 | **0.73** | 0.27 |
| people smiling | 0.012 | **0.71** | 0.29 |
| 1 person | 0.008 | **0.68** | 0.32 |
| smiling closeup | 0.007 | **0.87** | 0.13 |
| person smiling | 0.006 | **0.77** | 0.23 |
| person closeup | 0.005 | **0.84** | 0.16 |
| beard | 0.004 | 0.23 | **0.77** |
| car | 0.004 | 0.25 | **0.75** |
| selfie | 0.003 | **0.81** | 0.19 |

(b)

| *word/emoji* | *MI* | *p(female—word/emoji)* | *p(male—word/emoji)* |
|---|---|---|---|
| 😍 | 0.043 | **0.84** | 0.16 |
| ❤️ | 0.025 | **0.82** | 0.18 |
| 🔥 | 0.014 | **0.87** | 0.13 |
| 😘 | 0.007 | **0.79** | 0.21 |
| 💕 | 0.007 | **0.83** | 0.17 |
| beautiful | 0.006 | **0.90** | 0.10 |
| 💪 | 0.003 | 0.36 | **0.64** |
| bro | 0.003 | 0.05 | **0.95** |
| 🥰 | 0.002 | **0.85** | 0.15 |
| gorgeous | 0.002 | 0.32 | **0.68** |

(c)

| *alt_text+word* | *MI* | *p(female—alt_text +word)* | *p(male—alt_text+word)* |
|---|---|---|---|
| 1 person, beautiful | 0.042 | **0.94** | 0.06 |
| closeup, beautiful | 0.030 | **0.92** | 0.08 |
| 1 person gorgeous | 0.018 | 0.39 | **0.61** |
| smiling, beautiful | 0.017 | **0.87** | 0.12 |
| 1 person, pretty | 0.016 | **0.89** | 0.11 |
| closeup, gorgeous | 0.011 | 0.42 | **0.58** |
| closeup, pretty | 0.011 | **0.90** | 0.10 |
| smiling, pretty | 0.007 | **0.89** | 0.11 |
| selfie, beautiful | 0.006 | **0.85** | 0.15 |
| 1 person, cute | 0.005 | **0.80** | 0.20 |

(d)

| *alt_text+emoji* | *MI* | *p(female—alt_text+emoji)* | *p(male—alt_text+emoji)* |
|---|---|---|---|
| 1 person, 😍 | 0.037 | **0.80** | 0.20 |
| 1 person, ❤️ | 0.032 | **0.81** | 0.17 |
| closeup, 😍 | 0.022 | **0.88** | 0.12 |
| closeup, ❤️ | 0.018 | **0.83** | 0.17 |
| 1 person, 😘 | 0.015 | **0.89** | 0.11 |
| 1 person, 💕 | 0.013 | **0.80** | 0.20 |
| 1 person, 🔥 | 0.010 | **0.90** | 0.10 |
| smiling, 😍 | 0.009 | **0.87** | 0.13 |
| beard, 😻 | 0.005 | 0.13 | **0.87** |
| beard, ❤️ | 0.001 | 0.30 | **0.70** |

Table 1: MI result: (a) alt-text, (b) words/emojis, (c) correlation of alt-text and words, (d) correlation of alt-text and emoji.

to distinguish females from males. The matrix records co-occurrences of words, emojis, and tags in the same picture, not necessarily in a given window size $n$. We drop rare co-occurrence pairs that appear less than *50* times in total. In that way, we collect *2103* features from all the possible combinations of words/emojis, and alt-text in our data set. Table 1(c), and (d) take into account the correlation of generated alt-text with received words and emojis, respectively. We also compute the probability for a person to be male or female, given picture alt-text, and received words/emojis. In total, we select *4900* features from the above categories. After choosing these features, we apply feature extraction algorithms to downsample the features and retain only the ones that contribute the most to gender prediction. We evaluated individual and combined feature extraction methods to derive the best features set [2]. We apply these methods to find the best feature set $W_{best}$.

### 5.4 Retrofitting Words/Emojis Vectors

After selecting the best feature set, we compute vector representations of these features to evaluate the similarity of the online collected words from the target profile (which may contain new words or sequences of words) to our best feature set. To that end we use *word2vec* and *emoji2vec*. Our goal is to create a set of embeddings that accounts for both our offline collected dataset, *OCD*, and original word/emoji representations learned from *WE2V*. *Retrofitting* [12] is a process that adjusts an original word vector separately using a knowledge graph (e.g. *WordNet* [19]), in our case *OCD* instead. Retrofitting has advantages of being (i) a post-processing operation that does not require to browse the corpus again, (ii) applicable to any vector model, and (iii) simple and fast to implement. Retrofitting computes a new vector $v_i$ for the word/emoji $w_i \in W_{best}$, with the objective of being close to $w_i$'s original vector $v'_i$, when it exists, and also to vectors $v_j$ representing $w_j$ that are the $w_i$'s nearest overlapped words/emojis in *WE2V* or *OCD*. For that, we try to minimize this objective function:

$$\sum_{i=1}^{n} \left[ \alpha_i \|v_i - v'_i\|^2 + \sum_{j:w_j \in WE2V} \gamma_{ij} \|v_i - v_j\|^2 + \sum_{j:w_j \in OCD} \beta_{ij} \|v_i - v_j\|^2 \right] \quad (3)$$

We set $\alpha_i = 1$ when $w_i \in$ *WE2V* and *0* otherwise. An *overlapped* word/emoji belongs by definition to both *OCD* and *WE2V* datasets. The distance between a pair of vectors is defined to be the Euclidean distance. For $w_i$ in *WE2V* $\setminus$ *OCD* we take $\beta_{ij} = 0$ and $\gamma_{ij}$ is the *Cosine Similarity* score between $v_i$ and nearest overlapped words/emojis vectors $v_j$ in *WE2V* dataset. Cosine similarity is widely used to measure the similarity between two non-zero vectors by measuring the cosine of angle between them. For $w_i$ in *OCD*, we take $\gamma_{ij} = 0$ and $\beta_{ij}$ is the *Pointwise Mutual Information* (*PMI*) score [9] between $w_i$, and overlapped co-occurring words $w_j$. *PMI* has been extensively used in the field of *NLP* to measure words closeness based on their co-occurrence probability. *PMI* is formulated as:

$$PMI(w_i, w_j) = \log \frac{p(w_i, w_j)}{p(w_i)p(w_j)}$$

where $p(w_i)$, and $p(w_j)$ represent the probabilities that a comment contains $w_i$, or $w_j$, and $p(w_i, w_j)$ represents the probability that a comment contains both $w_i$ and $w_j$). Therefore, we calculate the vector $v_i$ by taking the average of the nearest overlapped words/emojis vectors $v_j$, considering their cosine similarity $\gamma_{ij}$, or pointwise mutual information $\beta_{ij}$ score according to the cases as follows:

$$v_i = \frac{\sum_{j:w_j \in WE2V} \gamma_{ij} v_j + \sum_{j:w_j \in OCD} \beta_{ij} v_j + \alpha_i v_i'}{\sum_{j:w_j \in WE2V} \gamma_{ij} + \sum_{j:w_j \in OCD} \beta_{ij} + \alpha_i} \tag{4}$$

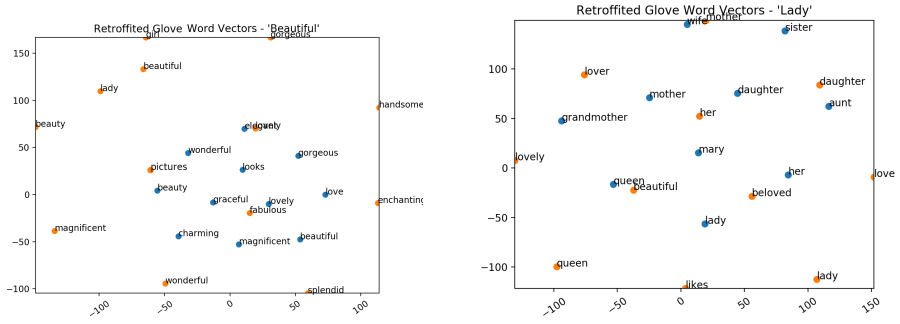
The advantage of adjusting the pre-trained words/emojis vector by using offline extracted data co-occurrences is two-fold: (i) handling non-overlapped words/emojis easily, and (ii) using sophisticated distributional embeddings (*WE2V*) that make the retrofitted vectors robust and suitable for gender inference attack process. For the feasibility of the computation, we truncate each sum in Equation 4 by summing only the *10* most significative terms (corresponding to the closest words to $w_i$). In the case of having a sequence of words as the best feature, we first retrofit each word vector separately. Then we take the average of the vectors associated with the words in the sequence. [21]. For example, consider *beautiful lady* as a best feature, we first retrofit *beautiful* and *lady* separately. Next, we get a vector for *beautiful lady* by averaging the retrofitted vectors of *beautiful* and *lady*. Figure 3 illustrates the separate word retrofitting, where the blue dots are *word2vec* vectors, and orange dots are the retrofitted vectors.

## 6 Online attack and gender obfuscation

Using our offline knowledge, the online phase consists of classifying a target user with unknown gender to the male or female category. For the demonstration purpose, we assume the target user is given.

### 6.1 Pre-processing and N-grams

This step follows by reformulating the words to their normal form, as detailed in Subsection 5.2. Next, as $W_{best}$ contains both single and sequences of words, n-grams (up to 3-grams) permit to find new words or sequences of words in the online extracted comments.




(a) (b)

Fig. 3: Retrofitting : (a) Beautiful, (b) Lady

### 6.2 Word/Emoji Similarity

We count the online occurrences of words belonging to $W_{best}$ and set the value of $w_{best}$ feature, where $w_{best} \in W_{best}$ to its number of occurrence. Otherwise, we find the closest words to $w_{best}$ by using their vector representation and set the $w_{best}$ value to the number of occurrence of close words. For each new word $w_{new}$, we compute the closest $w_{best}$ as follows:

$$\underset{w_{new}, w_{best}}{\arg\max}\ cosine(w_{new}, w_{best})$$

We ignore the result if the cosine similarity value is less than *50%*. For sequence of words in $w_{new}$ we proceed as in Subsection 5.4. This mechanism of replacing a new word by its closest word in the best feature set allows one to handle out of vocabulary words.

### 6.3 Gender Classification

In this step, the attacker has to evaluate the trained machine learning algorithm in an online mode. Given a target user, the algorithm outputs (i) *female*, (ii) *male*, or (iii)*unknown*. The output depends on the prediction probability threshold. We set this threshold to be *0.70*. For example, the output is *female* if the algorithm gives prediction probability of *0.70* to *female*, and *0.30* to *male*. Moreover, the output is *unknown* if the algorithm prediction probability for *female* is *0.65*, and *0.55* for *male*. In our experiments (see Section 7), we present the result of *700* users as they are labelled *female*, or *male*. Although the threshold empirically derived from our dataset, it is an arbitrary choice to be adapted to other datasets. It helps to prevent inaccurate attacks due to a lack of input information. For example, if the user has only one picture with an alt-text.

### 6.4 Comment/Picture Filtering

We now describe the filtering option proposed by our system for protecting user privacy. First, we discuss comment filtering, and later we define the picture filtering option.

***Comments Filtering Option.*** It suggests users hide comments that contain sensitive words, or sequence of words that disclose picture owner gender. Facebook offers two comments filtering options. First, the users can set up a list of words, phrases, or emojis that they do not want to receive from commenters. Facebook hides matching comments entirely from the published photos. Second, the users can manually select the comments and make them invisible from photos. The advantage of hiding comment is that it is still visible to the commenter and his/her friends, which reduces tension between the commenter and the picture owner.

***Pictures Filtering Option.*** As a suggestion, the user can hide the picture if the generated alt-text or combination of alt-text and comments leak the picture owner gender. Facebook settings allow users to restrict the picture visibility. We

| Contribution? | Feature | Value |
|---|---|---|
| +2.155 | hair | 1.000 |
| +0.161 | sobeautiful | 1.000 |
| +0.144 | absolutely | 1.000 |
| +0.087 | stunning | 1.000 |
| +0.050 | beauty | 1.000 |

(a)

| Contribution? | Feature | Value |
|---|---|---|
| -0.130 | dad | 1.000 |
| -0.132 | boy | 1.000 |
| -0.152 | beautifulpicture | 0.000 |
| -0.204 | sopretty | 0.000 |
| -2.348 | <BIAS> | 1.000 |

(b)

Fig. 4: Lime output : (a) Most contributed features (b) Least contributed features

use *LIME* [23] to alert the user about comments, or pictures that reveal owner gender and require to be hidden. *LIME* computes an explanation as a list of features (with weights) that either contribute to the prediction (in green) or are evidence against it (in red). Figure 4(a) presents the most contributing features, while, Figure 4(b) shows the least contributing features for a user, labelled by *female*.

As for gender obfuscation, the user can select some most contributing features as Figure 4(a), according to his/her desires, and then hides the comments, or pictures containing those features by following the above steps. Next, the user can re-run our online system component to check the gender inference attack vulnerability. The user can repeat the process until the output is *unknown*.

## 7 Experiments

In this section, we evaluate our approach for all three scenarios and demonstrate offline and online experiments.

***Offline Experiments.*** Using a Python crawler, we have randomly collected *627,776* pictures and their *1,332,219* comments. Facebook was unable to generate alt-text for *24833* pictures. We have kept those pictures for our second attack scenario, where we rely only on words/emojis usage for commenting pictures. The experiments are achieved by applying the classifiers from Python library *scikit-learn*. For result robustness, we apply several supervised machine learning

algorithms such as *Logistic Regression*, *Random Forests*, *K-Nearest Neighbors*, *Naive Bayes* and *Decision Tree*. To evaluate the classifier, we select the same number of males and females to prevent bias classification. Train-test splitting was preferable in this study as it runs k-times faster than k-fold. We choose the train-test size to be *70-30*, which gives the best accuracy. To address the problem of fairly estimating the performance of each classifier, we set aside a validation dataset. We train and adjust the hyper-parameters to optimize the performance of classifiers by using this dataset. Eventually, we evaluate the classifiers on the test dataset. Considering the extracted gender as the ground-truth, to evaluate our attack, we compute the *AUC-ROC* curve. The *AUC-ROC* curve is a performance measurement for classification problems at various threshold settings. In Figure 5, we show the *AUC-ROC* results for all three scenarios. In the first scenario, we rely on alt-text to infer the picture owner's gender. Figure 5(a) displays trained algorithms results on the extracted alt-texts features. Based on that, our trained algorithms can infer the target user gender with an *AUC* of *87%*. In the second scenario, we conduct an inference attack by training algorithms only on commenters' comments. As illustrated in Figure 5(b), the performance increases to *90% AUC*. In the third scenario, we train the classifiers by using the co-occurrence of alt-texts and words/emojis. Based on the Figure 5(c), *Logistic Regression* model which had *87% AUC* in the first scenario, and *90%* in the second scenario gets *5%*, and *2% AUC* boost in this scenario, respectively, which is a fairly substantial gain in performance.

To conclude, *Logistic Regression* performs the best in all scenarios. It is a discriminative model that is appropriate when the dependent variable is binary (i.e., has two classes). The results confirm our hypothesis that gender and picture contents have an impact on Facebook users' emotional responses. As a result, an attacker can train standard classifiers by using pictures metadata contained (i) other Facebook users' words/emojis preferences, and (ii) generated alt-text to infer the picture owner's gender. Note, as we rely solely on non-target generated data, the results cannot be compared to previous works that exploit data published by the target.

***Online Experiments.*** We have applied our online experiment to *700* users with their *21,713* pictures and their *64,940* corresponding comments. We have evaluated the performance of each classifier with *AUC-ROC*. As illustrated in Figure 5 (d,e, and f), *Logistic Regression* outperforms other classifiers in all three scenarios. Notably, the combination of alt-text and words/emojis boosts the performance of the classifiers in comparison to the other scenarios. To sum up, *Logistic Regression* is a suitable classifier for this task that can be trained by an attacker to perform a gender inference attack in online mode.

## 8 Discussion

The best scenario for the attacker is, as expected, the third scenario when he has access to Facebook generated alt-text and commenters' comments. The attack may work even when the target publishes nothing else than pictures. By applying our system to his published pictures, a Facebook user can check if he/she is

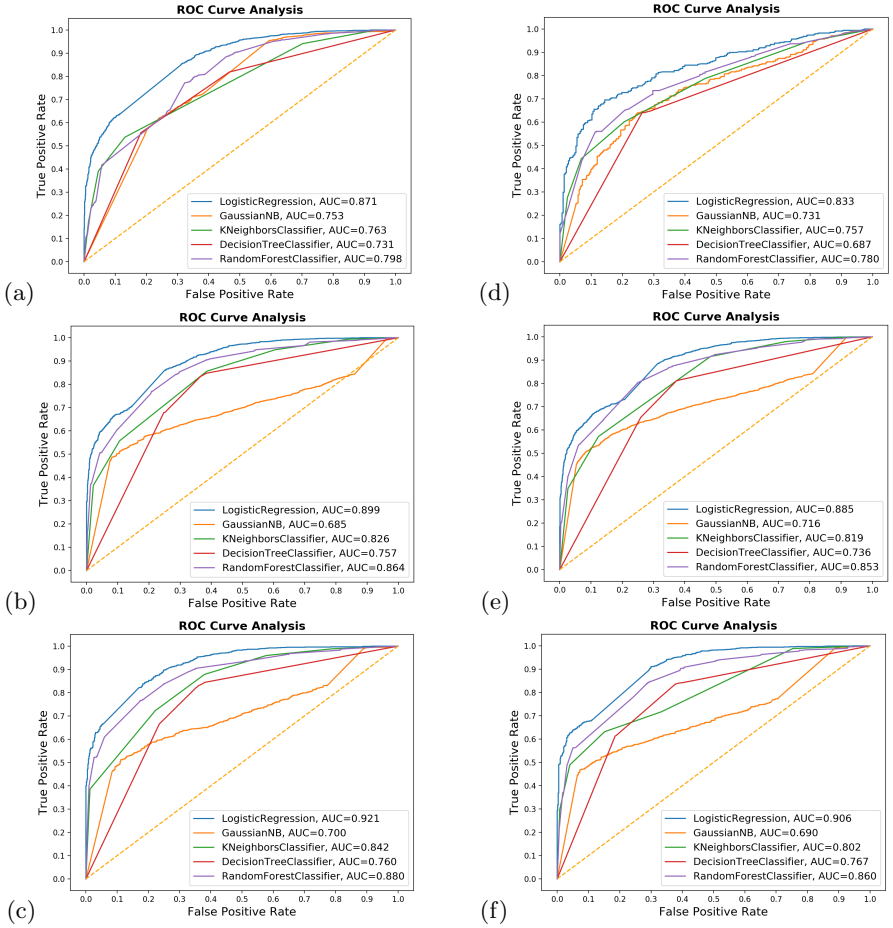



Fig. 5: AUC result of logistic regression trained on: (a) only alt-text (b) only commenters reactions (c) both alt-text and commenters reactions (d) removing alt-text features (e) removing commenters reactions features (f) removing alt-text and commenters reactions features.

vulnerable to gender inference attack of the above type. To counter the aforementioned privacy violations, we offer two countermeasures, namely: (i) hide some comments, or (ii) hide some pictures when they strongly contribute to the attack as explained in Subsection 6.4.

## 9 Conclusion

Identifying users' gender from their online activities and data sharing behavior is an important topic in the growing research field of social networks. It provides an opportunity for targeted advertising, profile customization, or privacy attacks. This study has investigated *627,776* pictures and their *1,332,219* comments. Based on the intensive analysis of the shared images, this work has demonstrated

(i) a new perspective of gender inference attack on Facebook users by relying on non-user generated data, and (ii) a privacy protection system. We have shown the possibility of gender inference attack even when all user attributes/activities such as profile attributes, friend list, liked pages, and joined groups are hidden.

As future work, we plan to use sophisticated algorithms to take the output of *LIME* and give the user the exact comments or pictures to be hidden in order to be obfuscated. We also plan to extend the current work to deal with the online inference of other attributes (e.g., age) and to explore online inference in other social network platforms (e.g. Twitter, Instagram). We may also take advantage of combining several user-generated content from different online social networks to infer private attributes.

## References

1. Chaabane Abdelberi, Gergely Ács, and Mohamed Ali Kâafar. You are What you Like! Information Leakage Through Users' Interests. In *19th Annual Network and Distributed System Security Symposium, NDSS*, San Diego, California, USA, 2012. The Internet Society.
2. Bizhan Alipour, Abdessamad Imine, and Michaël Rusinowitch. Gender Inference for Facebook Picture Owners. In *International Conference on Trust, Privacy and Security in Digital Business, TrustBus*, pages 145–160, Linz, Austria, 2019. Springer.
3. Bizhan Alipour, Abdessamad Imine, and Michaël Rusinowitch. You are What Emojis Say About your Pictures: Language-independent Gender Inference Attack on Facebook. In *The 35th ACM/SIGAPP Symposium On Applied Computing*, page 1826–1834, Brno, Czech Republic, 2020. ACM.
4. Tena Belinic. Personality Profile of Social Media Users How to Get Maximum from It. `https://medium.com/krakensystems-blog/personality-profile-of-social-media-users-how-to-get-maximum-from-it-5e8b803efb30`, April 2009.
5. Carole Cadwalladr and Emma Graham Harrison. How cambridge analytica turned facebook 'likes' into a lucrative political tool. `https://www.theguardian.com/technology/2018/mar/17/facebook-cambridge-analytica-kogan-data-algorithm`, May 2018.
6. Zhenpeng Chen, Xuan Lu, Wei Ai, Huoran Li, Qiaozhu Mei, and Xuanzhe Liu. Through a Gender Lens: Learning Usage Patterns of Emojis from Large-scale Android Users. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW*, pages 763–772, Lyon, France, 2018. ACM.
7. Ming Cheung and James She. An Analytic System for User Gender Identification Through User Shared Images. *TOMCCAP*, 13(3):30, 2017.
8. Munmun De Choudhury, Sanket S. Sharma, Tomaz Logar, Wouter Eekhout, and René Clausen Nielsen. Gender and Cross-cultural Differences in Social Media Disclosures of Mental Illness. In *Proceedings of the Conference on Computer Supported Cooperative Work and Social Computing, CSCW*, pages 353–369, Portland, OR, USA, 2017. ACM.
9. Kenneth Ward Church and Patrick Hanks. Word Association Norms, Mutual Information, and Lexicography. *Computational linguistics*, 16(1):22–29, 1990.
10. Ben Eisner, Tim Rocktäschel, Isabelle Augenstein, Matko Bošnjak, and Sebastian Riedel. emoji2vec: Learning Emoji Representations from their Description. *arXiv preprint arXiv:1609.08359*, 2016.

11. Reza Farahbakhsh, Xiao Han, Ángel Cuevas, and Noël Crespi. Analysis of Publicly Disclosed Information in Facebook Profiles. *CoRR*, abs/1705.00515, 2017.
12. Manaal Faruqui, Jesse Dodge, Sujay K Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. Retrofitting Word Vectors to Semantic Lexicons. *arXiv preprint arXiv:1411.4166*, 2014.
13. Lucie Flekova, Jordan Carpenter, Salvatore Giorgi, Lyle H. Ungar, and Daniel Preotiuc-Pietro. Analyzing Biases in Human Perception of User Age and Gender from Text. In *Proceedings of the 54th Annual Meeting of the Association for Computational*, pages 843–854, Berlin, Germany, 2016. ACL.
14. Neil Zhenqiang Gong and Bin Liu. You are Who you Know and How you Behave: Attribute Inference Attacks via Users' Social Friends and Behaviors. In *25th Security Symposium*, pages 979–995, Austin, TX, USA, 2016. USENIX.
15. Fariba Karimi, Claudia Wagner, Florian Lemmerich, Mohsen Jadidi, and Markus Strohmaier. Inferring Gender from Names on the Web: A Comparative Evaluation of Gender Detection Methods. In *Proceedings of the 25th International Conference on World Wide Web, WWW*, pages 53–54, Montreal, Canada, 2016. ACM.
16. Abbey Lenton. Facebook Wants you to Search Photos of your Female Friends at the Beach, but not your Male Mates. `https://www.whimn.com.au/talk/people/facebook-wants-you-to-search-photos-of-your-female-friends-at-the-beach-but-not-your-male-mates/news-story/bbc21ee6883bd07bfbbbe76a0c8ca54c`, February 2019.
17. Puneet Singh Ludu. Inferring Gender of a Twitter User Using Celebrities it Follows. *CoRR*, abs/1405.6667, 2014.
18. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*, 2013.
19. George A Miller. WordNet: a Lexical Database for English. *Communications of the ACM*, 38(11):39–41, 1995.
20. Dong Nguyen, Dolf Trieschnigg, A. Seza Dogruöz, Rilana Gravel, Mariët Theune, Theo Meder, and Franciska de Jong. Why Gender and Age Prediction from Tweets is Hard: Lessons from a Crowdsourcing Experiment. In *25th International Conference on Computational Linguistics, Proceedings of the Conference, COLING*, pages 1950–1961, Dublin, Ireland, 2014. ACL.
21. Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. Unsupervised Learning of Sentence Embeddings Using Compositional N-gram Features. *arXiv preprint arXiv:1703.02507*, 2017.
22. Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online Learning of Social Representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, 2014.
23. Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust you?" Explaining the Predictions of any Classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.
24. Lucía Santamaría and Helena Mihaljevic. Comparison and Benchmark of Name-to-gender Inference Services. *PeerJ Computer Science*, 2018.
25. Maarten Sap, Gregory J. Park, Johannes C. Eichstaedt, Margaret L. Kern, David Stillwell, Michal Kosinski, Lyle H. Ungar, and H. Andrew Schwartz. Developing Age and Gender Predictive Lexica over Social Media. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 1146–1151, Doha, Qatar, 2014. ACL.