An Empirical Investigation of Privacy via Obfuscation in Social Networks

By

Nick Reynolds
BIT Macquarie University

A thesis submitted to Macquarie University
for the degree of
Master of Research
Department of Computing
July 2019

MACQUARIE
University
SYDNEY·AUSTRALIA

Typeset in LaTeX 2_ε.

Declaration

I certify that the work in this thesis entitled AN EMPIRICAL INVESTIGATION OF PRIVACY VIA OBFUSCATION IN SOCIAL NETWORKS has not previously been submitted for a degree nor has it been submitted as part of the requirements for a degree to any other university or institution other than Macquarie University. I also certify that the thesis is an original piece of research and it has been written by me. Any help and assistance that I have received in my research work and the preparation of the thesis itself have been appropriately acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Ethics application reference number 5201800389 from the Faculty of Science and Engineering Human Research Ethics Sub-Committee.

Nick Reynolds

Abstract

Large quantities of personal profile information are available on online social networks like Facebook. This profile information can be used by an attacker to uncover a user's private attributes; in response to this, previous researchers have demonstrated how to obfuscate user profiles to reduce this attack vector. However, existing research has not yet examined the combination of network structure with profile information in this context, and the effectiveness of obfuscation techniques against that. Moreover, previous work examined the case of balanced private attribute classes like gender; inference of imbalanced classes — such as sexual orientation, which has been examined in the literature — poses additional challenges. This thesis examines these issues.

We found that previous obfuscation methods were less effective in reducing inference accuracy and in some cases not effective at all when an attacker used a combination of profile and network vectors. Extending obfuscation strategies to network structure could reduce the accuracy significantly, with just 20% obfuscation resulting in a drop in accuracy from 80% to 35%.

Unlike for balanced private attribute classes, the accuracy metric produces misleading results for imbalanced classes such as sexual orientation, where the F1 measure is more suitable. We show that there is a slightly higher risk of profile- plus network-based inference in this case and that network info is particularly useful here, in line with previous work, and show that obfuscation is required on both the network and profile side to reduce F1 for the positive class by half.

Contents

1 Introduction

Data privacy in online social networks (OSNs) has recently become a hotly debated topic, even a company so much as updating their privacy policy can receive large media attention and potential community backlash. Despite this, hundreds of millions of people share details about themselves on social media,[1] which in turn could be used by advertisers to target better ads, governments to spy on their people or individuals for any number of controversial tasks. At the same time, institutions and governments are releasing datasets to increase transparency and aid research, which has resulted in potential privacy issues. A well-known instance of this occurred when Netflix released an anonymised viewer dataset to the public with the goal of sourcing a better recommendation engine, unfortunately a statistical de-anonymisation attack was able to identify known users and discover their political views and sexual orientation [1].

More recently Australia's Department of Health released a dataset that incidentally exposed details such as whether individuals "take HIV medication, have terminated a pregnancy, or is seeing a psychologist"[2].[2] The Department of Health had de-identified the data, but this could not prevent 'attackers' linking it to other sources of information to determine sensitive attributes about prominent Australian Members of Parliament and a footballer. Another academic instance involved the use of de-identified credit card metadata to re-identify people with a 90% success rate [3].

In the social media space, using real data, Chen et al. [4] have shown how public Facebook profile information can be used to infer private profile attributes (Figure 1.1) (gender, relationship status, age, interested_in). They implemented a range of machine learners which used a profile's publicly listed movies to predict their gender with an accuracy greater than 80% against a baseline of approximately 57%.

To reduce the effectiveness of profile inference Chen et al. [4] evaluated a number of obfuscation techniques that could be applied to the public attributes (Figure 1.1). There were two dimensions to this obfuscation, the 'policy' of whether to add, remove or replace attributes and the 'strategy' used to select the order in which the attributes had the policy applied. They tested four strategies: χ^2 (a statistical measure of the strength of association between each attribute and a target), majority features, most popular features and random. They found that χ^2 was the best strategy with add and replace policies being the standouts.

Other research [5–10] has shown how the structure of an OSN can also be used to infer these private attributes. Much of this research uses Twitter or other social network data, which does not have the richness of Facebook's profile information. In this thesis, we explore the use of Facebook network data to infer a user's attributes following with Chen et al's approach, and then design and

[1] Facebook Company Info https://newsroom.fb.com/company-info/

[2] Health record details exposed as 'de-identification' of data fails https://www.smh.com.au/technology/australians-health-records-unwittingly-exposed-20171218-p4yxt2.htm

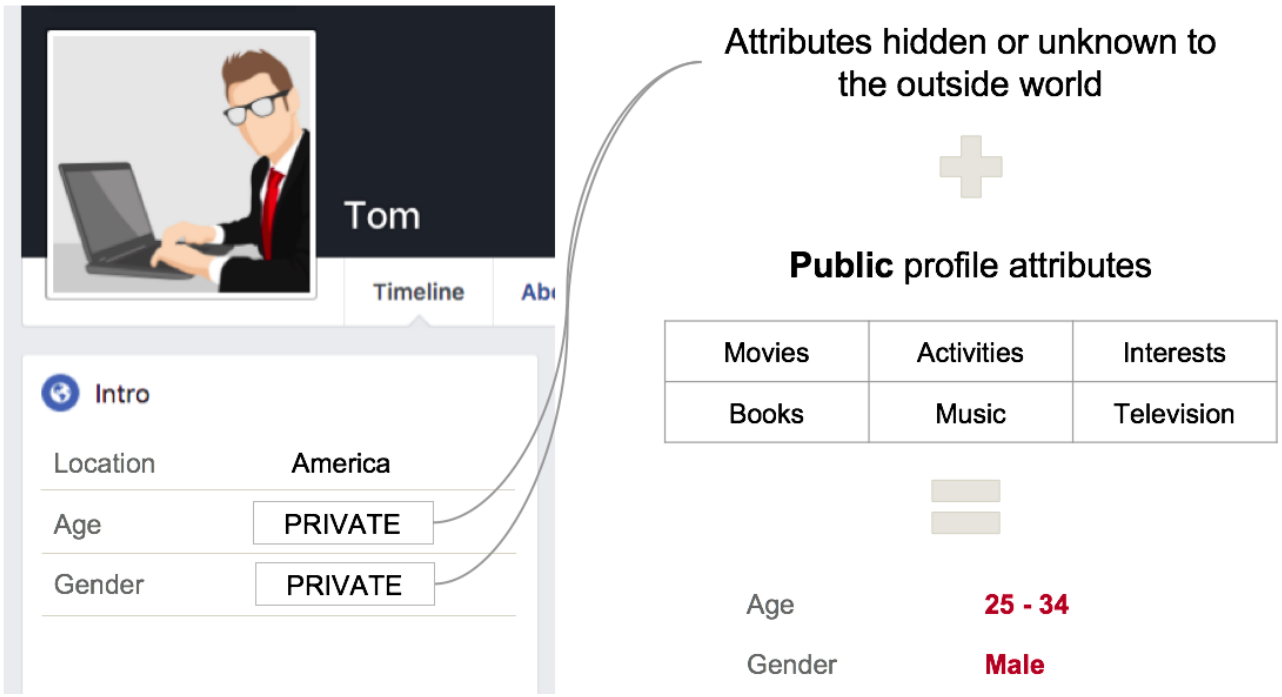

FIGURE 1.1: Chen et al. Facebook Profile Inference

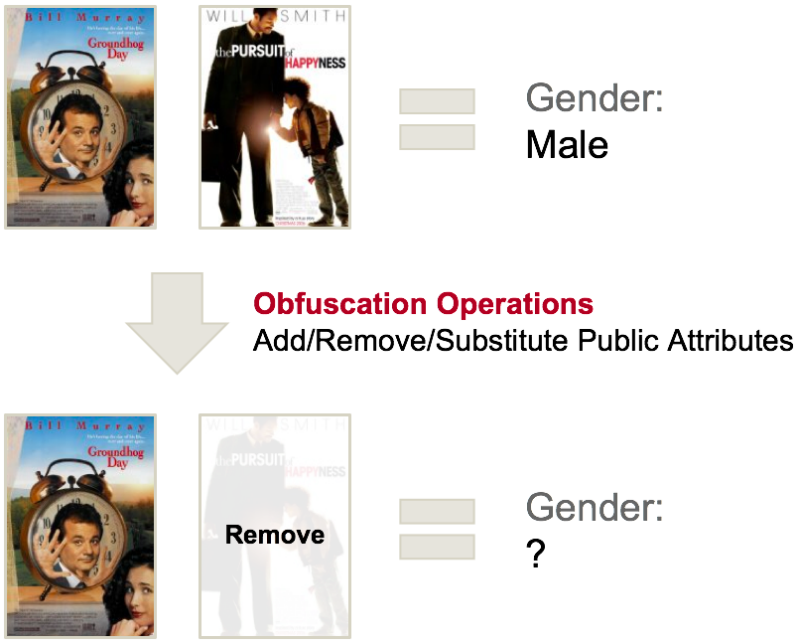

FIGURE 1.2: Chen et al. Facebook Profile Obufscation Overview

	C1	C2	
C1	35	15	50
C2	10	40	50

Accuracy = 0.75

	C1	C2	
C1	37	13	50
C2	15	35	50

Accuracy = 0.72

TABLE 1.1: Balanced Classes: rows are actual classes, columns are predictions. Before (left) and after (right) obfuscation

	C1	C2	
C1	1	4	5
C2	21	74	95

Accuracy = 0.75

	C1	C2	
C1	3	2	5
C2	26	69	95

Accuracy = 0.72

TABLE 1.2: Imbalanced Classes: rows are actual classes, columns are predictions. Before (left) and after (right) obfuscation

implement obfuscation approaches against such inference. These approaches can be combined with profile inference and obfuscation techniques creating real world scenarios where an attacker has obtained both vectors and obfuscating against such an attack. The research questions we focus on for this component of the work are thus as follows:

- Q1a: How well does network-based inference work on Facebook data, and what obfuscation techniques work against it?
- Q1b: What's the effect of profile-based obfuscation on profile+network (P+N) inference?
- Q1c: How can obfuscation be extended against P+N inference?

We then extend [4] in a second way. Class imbalance is a property of a dataset that can have substantial implications for machine learning metrics [11]. Accuracy is the default metric for classification. In the constructed example of Table 1.1 we can see that the accuracy of the first table is 75% $((35+40)/(50+50))$; the second table could represent a small amount of obfuscation such that the accuracy is reduced to 72%, a 3 percentage point difference, with each class experiencing only small changes. However, accuracy becomes misleading where imbalanced classes are involved. In Table 1.2 we see that C1 only has a total of 5 items, versus C2 having 95. After obfuscation on this set we can see that overall accuracy has decreased 3 percentage points, with C2 correct predictions going from 74 to 69, however this masks the change in the C1 class, which has an increased number of correct predictions from 1 to 3 (20% to 60% of cases). This is especially important if the small class C1 represents a particularly sensitive attribute.

A real-world instance of such a class imbalance that has been discussed in the literature [12, 13] is sexual orientation, where individuals in the minority class form a very small subset of the greater population. This specific case (which we can derive from our data, as a combination of gender and interested_in) will be explored in the later part of this thesis: we investigate how inference and obfuscation work under a more appropriate metric, the F-measure.

- Q2a: What's the effect of profile-only inference and obfuscation under a metric reflecting class imbalance?
- Q2b: And for P+N inference?
- Q2c: What is the most appropriate obfuscation technique here?

We begin with Chapter 2 discussing some of the literature concerned with profile and network inference, as well as obfuscation approaches and techniques. Chapter 3 will focus on replicating the inference and obfuscation work done by Chen et al. [4], while Chapter 4 extends this work to include inference based on network structure, alone and in combination with profile-based inference, and obfuscation against those. Chapter 5 experiments with the handling of an imbalanced dataset, with our class instance being sexual orientation from the Chen et al. dataset.

2 Literature Review

It is hard to avoid social media in 2017 as some of the top visited sites online are social networks with the biggest of these being Facebook, YouTube, Instagram and Twitter.

- **Facebook (2014)** [1] has the highest number of active users daily (1.37b) and monthly (2.07b). Users can make their profiles reflect their various interests and 'friend' other users.
- **YouTube (2005)** [2] is primarily a video sharing website, where users can subscribe to each other to view future postings. They have around 1.5b monthly users.
- **Instagram (2010)** [3] is a photo sharing website where users connect through following each other. They have 700m users active monthly and 500m daily.
- **Twitter (2006)** [4] allows users to post small 'tweets' with a limited number of characters. Users connect by following each other. They have 319 million active users.

Most inference technique work has been done using Facebook or Twitter social network data, likely due to the communication being largely text based interactions which is significantly easier to process than videos on YouTube or photos on Instagram.

2.1 Recent Work on Single Profile Inference

Single profile inference involves using a profile in isolation, not taking into account the 'friends' or network aspect of the profile. There are a number of different attributes that can be predicted about a person by their profile. Here we present some examples that are fairly typical in their use of machine learning for inferring these attributes.

Gender, Age, Regional Origin and Political Orientation (2010) Identifying that inference about a persons demographics can have implications in advertising and personalisation, Rao et al. [10] sought to use Twitter data to predict these 'latent' attributes in a "first-of-a-kind application". They were able to draw on some of the authors' previous work stemming from extracting similar attributes from telephone conversation and email transcripts using sociolingistic features, noting that e.g. "women 'LOL', men tend to 'LMFAO'" [14]. Data was extracted from Twitter via a breadth first search from a random seed user, excluding those with a high follower count as these were deemed to be celebrities. For gender they extracted 500 users for each class with 405,151 tweets, this also included features such as the number of followers / following users and the

[1] Facebook Company Info https://newsroom.fb.com/company-info/

[2] YouTube has 1.5 billion logged-in monthly users watching a ton of mobile video http://tcrn.ch/2svWSl7

[3] Instagram by the Numbers: Stats, Demographics & Fun Facts https://www.omnicoreagency.com/instagram-statistics/

[4] Twitter is now losing users in the U.S. http://money.cnn.com/2017/07/27/technology/business/twitter-earnings/index.html

response / retweet rate for their tweets. The number of users in each gender class was even making the chance baseline 50%. These were fed into their stacked Support Vector Machine (SVM) achieving a result of 72.33%. They noted that interestingly women are 3.5 times more likely to use emotes, 1.5 times more likely to use ellipses and 1.4 times more likely to repeat characters e.g. 'NO WAAY'. Interestingly, targeted advertising can also be used on Facebook to expose similar features about a user shown by Korolova in 2010 [15] and more recently Faizullabhoy et al. in 2018 [16].

Gender, Sexual Orientation, Race and Political Ideology (2012) Kosinski et al. [12] used a sample of 58,466 users from the United States who volunteered their Facebook data by completing a personality test on 'myPersonality', a site which is no longer available. The data collected through the test included their 'likes', demographic profiles and various psychometric test results. They used logistic and linear regression classifiers attempting to predict a wide range of results: "sexual orientation, ethnic origin, political views, religion, personality, intelligence, satisfaction with life, substance use ... and basic demographic attributes such as age, gender, relationship status, and size and density of the friendship network." [12] For gender their set had 62% labelled female and for orientation they had 4.3% of males labelled as gay and 2.4% of females as lesbian. Area Under Curve (AUC) is the metric they use to evaluate their models—they are able to achieve an AUC of 93% for gender (50% AUC as their baseline), 88% for sexual orientation, 95% for race and 85% for political ideology. As this data was volunteered by users, it is in some ways richer than one collected from a broader random group of users. The authors include a caveat in their work: "One can imagine situations in which such predictions, even if incorrect, could pose a threat to an individual's well-being, freedom, or even life."

Gender, Age, Relationship Status and Interested In (2014) Chen et al. [4] used a dataset of 1.9 million profiles in their work where they inferred a profile's private attributes (age, gender, interested in and relationship status) using the profiles public information (movies, music, activities), it is not mentioned how this dataset was procured. They empirically compared three classification techniques: Logistic Regression, Naive Bayes and Random Forest showing that a profiles gender is the least challenging to predict with an accuracy of up to 85% while age was the most difficult with 62%, although no baselines were provided. Their conclusion was that Naive Bayes was the most effective inference technique.

Political Ideology (2017) Preotiuc-Pietro D. et al. [17] demonstrated that inference using a single profile was still viable last year when they set out to predict the US political ideology of Twitter users. United States Twitter users were recruited through a Qualtrics form to fill out demographics (incl. gender and political ideology) about themselves and to provide their Twitter handle. 3,938 users participated and were compensated with 3 USD for completing the 15 minute survey. Users described their political ideology on a scale of 1-7, 1 being very conservative, 7 being very liberal, participating users then had their tweets extracted into the dataset. They used a number of linguistic features:

- Word2Vec [5] [18] is used extensively in natural language research, it provides a group of

[5] https://code.google.com/archive/p/word2vec/

models that use a two layer neural network to map words to vectors for processing.
- Unigrams, a bag of words representation of a users tweet history.
- Linguistic Inquiry and Word Count (LIWC) typically used for psychological studies.
- Emoticon lexicons.

The emotion lexicons created an emotion for each tweet, which was averaged across all tweets to assign each user an emotion. Words were classified as political words, politician names and political media sources.

This corpus was then used for prediction into contrasting buckets (1 vs 7, 2 vs 6) where they compared their political terms clusters with linguistic clusters, finding that their political terms cluster outperforms Word2Vec at the binary classification task. They also attempted multi-class classification across all 7 steps in the sliding scale. They found that the maximum accuracy they could achieve was 27.6% using graph regularisation and a pre-computed political weighting table, their baseline was 19.6%.

Preotiuc-Pietro D. et al. first concluded that users in their dataset are far less likely to post about politics [17]. From their results they also note that political ideology placement is correlated to political engagement, "only self-reported extremists appear to devote much of their Twitter activity to politics at all." [17]

Splitting political ideology into a scale was an intelligent move, as it's rarely a binary classification problem but is usually treated as such in research [10]. Their methodology is well thought out, covering the fact that most users do not list their political ideology on Twitter, so instead of grabbing that subset of users (which would skew their results) they actively contacted users asking them to complete their questionnaire (for a nominal payout). They do note that their dataset is skewed towards the very liberal end of the spectrum which could affect the model though this doesn't seem to be mentioned throughout the rest of the paper. The key learning here is that text inference by itself can still be powerful, but as we continue to discuss this area in the next section we see that incorporating the social network aspect of Twitter may have improved the predictors accuracy.

2.2 Inference Using Social Network Structure

Interests (2008) Looking back now to 2008, we note that He and Chu [5] were early proponents of modelling a social network to infer data. They used a Bayesian network to model a simulated social network derived from Epinions.com data, a popular review website where users could 'follow' each other. Their inference attacks focused on determining a users interests, they achieved accuracy levels of 60.6%-76.5%, where they could most accurately predict users who were interested in electronics as it was the most popular interest.

We note that they did not have a true social network to use for their inference, relying on a review website which does not necessarily reflect the same relations as something like Facebook. It is not immediately apparent how this could be applied to a genuine OSN. However, their key point is still applicable to today's research: "inference accuracy increases as the influence strength increases between friends" [5].

Political Ideology (and others) (2014) Instead of treating inference as a static problem, Volkova et al. [6] looked at it as a streaming data problem, noting that users tweet fairly infrequently and that restrictions to Twitters API means that only a small number of tweets can be gathered for analysis. Using the Twitter API they collected lists of followers and friends, and extracted user mentions, hashtags, replies and retweets. Volkova et al. evaluated their streaming model with random dynamic Bayesian updates, cross validated using 70% to train, 10% for development and 20% for testing for each fold. Users friends Tweets were also incorporated as binary features, they note that this is to overcome sparsity issues, the number of communications per neighbour and the number of neighbours per interest. They determined that relatively small sets of 50-100 tweets can be used to classify a users political stance with close to 100% accuracy; this depends on how active the user is and the types of content they are sharing and who they are interacting with.

Sexual Orientation (2014) Sarigol et al. [13] wanted to demonstrate how a users online friends increase the risk of their private data being leaked, with their example being sexual orientation. They used a publicly available dataset from Friendster, a defunct Social Network as of 2015 [6], the dataset contained 3.4M public profiles and 11M social links. To incorporate the social network into their dataset they used a number of aggregated features from Table 3 [13] (note they use k 1, 2 and 3):

- Number of users at distance k
- Average age of friends at distance k
- Gender counts at distance k
- Relationship counts at distance k
- Romantic interest counts at distance k
- Sexual orientation counts at distance k
- Weighted frequency of friends of each sexual orientation

They showed that data leakage risk increases as friends share their own private information, for example homosexual males are at a higher risk of having their private information leaked compared to heterosexual males, when friends reveal their own sexual orientation. They also concluded that a user does not even need a profile in the social network to be compromised, if an attacker can gain a list of your friends or who you associate with on another platform / in person then they are still at risk.

Geolocation (2015) Geolocation is a real world location estimation for an object, in our case a user — this can used to refer to a specific latitude/longitude or a text based approximation i.e. "Sydney", "Macquarie University". Rahimi et al. [7] explored using label propagation (LP) and text inference with logistic regression (LR) to improve the inference of Twitter users' geolocations.

In recent work, label propagation (LP) has been given preference over older techniques for modelling social networks [7, 8]. LP is a technique applied to a graph structure, where each node will represent a user, and the edges connecting these nodes will represent their connections. Nodes should begin with a label, such as their gender or political affiliation, which will then be propagated throughout the graph based on a cost function. As a result nodes inherit new labels,

[6] http://www.friendster.com/

for example a node may now have a gender which was previously unknown or different to what was set originally.

In Rahimi et al. [7] they used LP across three different Twitter corpora:

- GeoText: $\sim 380K$ tweets from 9.5K users for contiguous USA. Users location is their latitude and longitude from their first tweet.
- Twitter-US: $\sim 39M$ tweets from 450K users for contiguous USA. Users location is their latitude and longitude from their first tweet.
- Twitter-World: $\sim 12M$ English tweets from 1.4M users worldwide. Users location is their latitude and longitude from their closest city.

They showed that on two of three three corpora a hybrid approach of LP-LR achieved higher accuracy levels than the two approaches applied separately and higher than their related work benchmarks. The LR method was more effective on the Twitter-World dataset than any other method due to the connected-ness of the test nodes. The experimental results support their claim that network based models are generally superior to text-based geolocation models and that their hybrid approach is novel and handles the problem of disconnected users in a graph.

Anti-Vaccine Sentiment (2015 & 2016) Zhou et al. [19] researched whether they could model anti-vaccine opinions regarding human papillomavirus (HPV) by looking at a sample of 42,533 tweets from 21,166 users consisting of 8,261 text fragments and 10,758 social connections (users they followed, and users who followed them). A sample set of tweets were extracted and tagged by the researchers, with the final sample consisting of 884 tweets in the training set and 907 in the testing set, of which 247 (28%) and 201 (22%) were labelled as anti-vaccine, respectively [19]. Pre-processing included removal of URLs, punctuation and "words that were unlikely to confer meaning".

The dataset was broken into two different sets, both consisting of three months worth of tweets; the first used as a training set and the second used as a testing set. They used an SVM with a radial basis function kernel, and computed accuracy, precision, recall and F1. Their most accurate classifier used only the network information and had an accuracy of 88.6%, their content-only classifier had an accuracy of 85.2% while a combination of the two scored 88.6% — the same as network by itself. While the accuracy results are similar they conclude that classifiers trained on pure network features often have higher precision and lower recall [19].

They do not report the final F1 scores for their results on the test period, which could be highly varied compared to the accuracy scores, as shown in their training results (Table 2) the network classifier that had the highest accuracy on the test set only has an F1 of 0.67. They do not concretely say how the network was incorporated into their dataset, whether it was binary features, counts or some other approach.

Ramezani et al. [8] extends upon [19] using LP to model the spread of anti-vaccine sentiment throughout the Twitter social network. They first implemented Zhou et al's approach [19] with the same dataset as a baseline then moved on to augmenting this dataset with additional features based on each tweets contents: emoticons, links and topics. A number of additional vertices were created for LP: user links, hashtags, emoticons, N-Grams, topic, URL, URL N-Grams and specific word vectors. Their new implementation used a two step approach, LP with a classifier (LR) which was more effective overall as their reimplementation approach achieved F-scores of 0.902 and

0.931 for the two periods, while their new implementation reached scores of 0.977 and 0.942. They largely attribute this to their extension of the tweets attributes.

Ramezani et al. [8] observed that all sources of information in a message can contribute to sentiment analysis inference, and hence included the messages emoticons and URLs, instead of stripping them out as [19] did. They also note that a user's entire network is relevant for carrying out inference, not just their direct connections, although they state that "approaches taken in the present work did not appear to fully take advantage of the extended label propagation model" [8]. Further, and relevant to the work of this thesis, they discuss why accuracy is not a good metric for their evaluation, and why F1-score should be used instead.

Social Behaviour Rating (2016) Gong and Liu [9] proposed an attack model aiming to infer a user's (university) major, employer, and cities lived. Their proposed "social-behaviour-attribute (SBA) network" considers social structure, user behaviours and user attributes when inferring attributes about a user, from this they propose the attack method 'vote distribution attack (VIAL)' to perform attribute inference. To back up their theoretical explanation they scraped data from two domains:

- Google+ ($\sim 1M$ users and $\sim 5M$ connections) for social network and profile attributes: major, employer and cities lived.
- Google Play ($\sim 48K$ items and $\sim 3M$ reviews) for users viewers on books, TV shows, apps etc.

They constructed various nodes out of this data: social nodes for individuals, behaviour nodes (reviewing an item) and attribute nodes (employer, major, cities lived). Each node was connected based on whether they had that attribute, had reviewed an item, or existed as friends. This created multiple attack vectors for inferring attributes nodes about an individual either via their behaviour nodes, social nodes or both.

Their VIAL method was benchmarked against other attack models on behaviour-only, social-only and combined approaches and was found to score better in F1, precision and recall. Gong and Liu [9] go into a substantial amount of detail, however their practical implementation centres around Google Plus — a social network with little real world usage compared to Facebook or Twitter [20]. Their approach could be applied in other domains provided the datasets can be tied back to Facebook or Twitter users.

Gun Violence (2017) Similar to [8, 19], Green et al. [21] modelled the 'spread' of gun violence as an epidemic spreading through social interactions; they hoped to predict individuals that would be the target of gun violence either fatal or non-fatal. Two datasets provided by the Chicago Police Department were used for this study:

- **Arrests** $\sim 1M$ arrests from January 1, 2006 and March 31, 2014, involving $\sim 462k$ people. This dataset contained incident codes as well as demographic information: birth date, race, sex, and gang membership.

- **Gunshot Crimes** $\sim 16K$ incidents ($\sim 84\%$ non-fatal) involving $\sim 14K$ people, 90.2% could be mapped to the arrests dataset.

The incident codes in the arrests dataset were used to create a network of 138K individuals with separate nodes for people and offences, with edges connecting people and events (noting that people cannot be connected directly to people, and events cannot be connected to events).

Their demographic and network modelling was done using a 'Bayesian Hawkes Process' [7], they found that models based on both demographics and social network structure were the most effective, predicting 728 of 1,382 people at the highest risk of being shot, while demographics by itself predicted 475 and network by itself predicted 589. This further supports the idea that both network and profile together can create a more effective model.

2.3 Online Social Profile Obfuscation

The works from He and Chu [5] described in Section 2.2 and Chen et al. [4] in Section 2.1 both considered obfuscation techniques after their inference attackers had been developed.

He and Chu [5] proposed four protection rules: hiding attributes (HA), falsifying attributes (FA), hiding relationships (HR) and adding relationships (AR). The main obfuscation benchmark (Figure 14.9 in [5]) measured the percentage of successful protection (i.e. the attribute was predicted incorrectly. The successful protection rate was plotted against the inheritance strength of users in the network. AR was the consistent leader with 0.38 successful protection, FA fluctuated from 0.08 to 0.38, HR and HA stayed lower with HR consistently above HA.

They concluded that adding relationships (AR) was the most effective protection rule, as it modifies the social graph in the greatest way. Falsifying an attribute will have a greater affect on a node than if the attribute didn't exist at all, hence FA > HR. Finally, hiding attributes (HA) is least effective as there are still clues throughout the social network to determine values.

Their obfuscation implementations assume that the attacker is using Bayesian inference (the same approach He and Chu used) and as a result the obfuscation may be less effective against different methods of inference.

A more general obfuscation approach was taken by Chen et al. [4], where they attempted to make very generic obfuscation approaches and test them against their aforementioned inference attackers. They used feature selection to determine which features should be changed in their various obfuscation policies. They began testing obfuscation strategies with four proposed methods for this feature selection: χ^2, majority, popularity and random. This was then combined with three different obfuscation policies: adding, removing or replacing features.

They introduced a variable for the rate to which obfuscation is applied (R), $R = 0$ being not at all. To evaluate these strategies they applied obfuscation to the dataset with various values of R and recording the classifiers accuracy results after the obfuscation had been applied. The accuracy for the different obfuscation strategies and policies was plotted in Figure 3 in Chen et al. [4].

χ^2 was found to be the most effective strategy while the add and replace policies were shown to be more effective than the feature removal strategy, this is to say that these approaches had the lowest accuracy results after obfuscation. In every approach in Figure 3 from Chen et al. [4] we see χ^2 out performing the other approaches. Obfuscation using rules proves far more effective than random approaches and falsifying attribute values proves more effective than hiding the attributes.

[7] See eMethods in supplemental content https://jamanetwork.com/journals/jamainternalmedicine/fullarticle/2594804

Chen note towards the end of their paper that if all users start obfuscating then the obfuscation loses its effect, and they also point out that users may not want to remove features from their profiles.

The authors further observe that alterations to a user's profile come with their own *cost*, changes need to align with user preferences otherwise the obfuscation suggestions (i.e. add these 4 movies) will be discarded and as a result privacy will not be protected. To further investigate these user preferences Chen et al. [4] crowdsourced responses using an online survey platform (where users already had profiles with demographic data) asking:

1. How sensitive is your profiles data? (1-5)

2. Rate the preference of adding a number of specific movies to your profile. The movies were selected by Chen et al. based on 3 factors: their IMDB rating "quality", χ^2 in their dataset "protection level" and count of users with this movie in their dataset "popularity".

There were 158 valid responses. Each user was classified as privacy cautious (Question 1 answer ≥ 3) or non-privacy aware based on their answers to the first question. They noted that users who are more security conscious have a higher likelihood of accepting higher "protection level" movies, they found that all respondents prefer movies that are high "quality" and high in "popularity". They perform a basic obfuscation method using this information, potential future work could involve building this out to include removal or replacement of features based on user preferences.

2.4 Summary

While there are a number of different approaches for profile and network inference, we will be specifically extending the work of Chen et al, both their inference and obfuscation techniques. Obfuscation techniques used in He and Chu [5] will become relevant when we move into obfuscating against network inference and will guide our approach.

3 Replicating Foundation

3.1 Introduction

This chapter will focus on the replication of key work from Chen et al [4] discussed in Chapter 2. The primary task tackled in their paper involved building classifiers from a user's publicly listed movies to infer/predict their (potentially private) gender attribute. We follow this same approach using the same features and target attribute. Chen et al. [4] proposed a number of obfuscation techniques, methods applied to a user's features to mask their private attributes usually by adding, removing or manipulating their features. We replicated these techniques achieving similar results.

3.2 Data Preparation

Personal Profile Data We obtained Facebook data extracts from the CSIRO Data 61 team who authored the paper, including a MySQL database `osn_obfuscation.sql` that contained Facebook profile information such as gender, date of birth, relationship status and interested in. It also included a number of features that a user can add to their profile, such as what movies they like, music they listen to or sports they play. Chen et al.[4] do not describe how the data was gathered, but presumably via some sampling strategy. We loaded this data into a MySQL instance running locally in Docker. [1]

As a first step we attempted to reconcile our profile summary with that of Chen et al. (Table 1 in [4]). In order to do this we needed to calculate age from date of birth and cleanse

[1] https://www.docker.com

Age: 35,058 users (1.2%)			**Gender**: 1,149,201 users (39.35%)		
Classes	Code	Percent	Classes	Code	Percent
13-17	0	0% (0)	Female	0	57.4% (659,644)
18-24	1	4.1% (1,436)	Male	1	42.6% (489,557)
25-34	2	63% (22,088)			
35+	3	32.9% (11,534)			
Relationship: 383,557 (15.12%)			**Interested In**: 319,014 users (10.92%)		
Classes	Code	Percent	Classes	Code	Percent
In a relationship	0	62.97% (241,529)	Men	1	23.89% (76,228)
Single	1	37.03% (142,028)	Women	2	39.32% (125,440)
			Men and Women	3	36.78% (117,346)

TABLE 3.1: Target attribute availability and their classes; total users: 2,920,439

relationship data. As DOB was a free-text field containing values such as "May 20 2010", "1st Dec" or "20/10/2003" a Python multilingual date parser was applied across these values with any undefined cases being treated as empty. The relationship data contained key phrases but mixed in with other terms such as "In A Relationship with Tim", there were also a number of these in other languages or different character encodings. A mapping was written to take values containing 'Single', 'Widowed', 'Divorced' and 'Separated' and encode them as 'Single', then to take 'Married', 'Complicated', 'Relationship' and 'Engaged' and encode them as 'In a Relationship'. A summary of the profile data can be seen in Table 3.1.

Our user base sits at $\sim 2.9M$ while the Chen et al. data summary (Table 1) only includes $\sim 249K$ users; from this it is apparent the dataset has been altered since the original publication. The age group distribution is fairly different, with our dataset showing most users in the 25-34 bucket while Chen had more users in the 18-24 bucket. The distributions for relationship, gender and interested in attributes are all practically identical to Chen's results. This suggests that the dataset had been subsequently updated, as it has been 3 years since its initial collection — these other attributes are more stable than age which naturally changes over time.

Profile Entertainment Data We observed that all entertainment data such as movie names were stored as strings. This means that there could be variant names (misspellings, typos) for the same movie for example: 'twilight: new moon', 'the twilight saga: new moon', 'new moon twilight' and 'twilight - new moon'. We therefore also investigated processing the data by considering strings within a certain Levenshtein distance to represent the same movie. Levenshtein distance (LD) is a measure of string similarity, with the resulting metric being the minimum number additions, deletions or replacements necessary to create the target string [2][3]. For example `Lauren` to `Laura` would have LD = 2, the first change is substituting the 'e' for an 'a' and second change is removing the 'n'.

To establish a canonical set of movies names, we downloaded an auxiliary movies dataset from The Movie Database (TMDB) [4]. This dataset contained the top 5,000 movies and was used with LD in an attempt to clean the movie list in the profile dataset. The output of this process was a table containing movies that matched on LD for values 0 (no change), 1 change and 2 changes; movies that were unmatched or had an LD outside this range were dropped from this set. Chen et al. only included users in their dataset that had at least 10 movies associated to their profile. We investigated several thresholds here.

3.3 Replicating Profile Inference

3.3.1 Method

In this section we will be predicting a user's gender using movies from their profiles, following the approach of Chen et al.

[2] http://www.levenshtein.net

[3] This is a very coarse method, there are other similar measures that could have been used but we only need a coarse notion of distance.

[4] https://www.kaggle.com/tmdb/tmdb-movie-metadata

Evaluation Framework As in Chen et al, we use stratified k-fold cross validation with 10 folds. Stratified k-folds creates training/testing subsets from a dataset by preserving the percentage of samples for each class. Cross validation attempts to avoid over-fitting a model to a dataset, without 'folds' a model training and testing on the same dataset will give great results but the model will be specifically trained for that set. Ten fold cross validation, allows us to train on 9 folds and test on the left out one — continuing this 10 times to create an average result of the metrics computed on each fold.

A tuning function was used to determine the best parameters for each estimator on one of the folds used specifically for this. The tuning function was set to optimise for accuracy and to compute both accuracy and AUC metrics. Accuracy is defined as the percentage of correct predictions out of the total number of predictions. Area Under the Receiver Operating Characteristic Curve (ROC AUC) plots a curve of the true positive rate against the false positive rate, the higher the area under this curve is then the higher the true positive rate is without increasing the false positive rate [22]. We used Python's `sklearn` for implementing all of this. [5]

Learners As we are replicating Chen et al. we selected the same classifiers: Naive Bayes, Logistic Regression and Random Forest. More detail is available in sources such as Alpaydin [22] or Hastie et al [23]. In addition we included a majority baseline classifier for comparison.

- **Naive Bayes (NB)** NB treats features as independent, assuming that they all contribute separately to the prediction, which is useful when you do not have enough data to calculate the dependency accurately. NB uses Bayes theorem to estimate the probability of an event based on features which may be relevant. To avoid NB assigning zero probability to categories that are not present in the training set, a smoothing parameter α for probabilities is used. We used a Bernoulli NB model for binary features.
  Hyperparameters:
  α: [0.0001, 0.001, 0.01, 0.1, 1] — Controls the amount of smoothing.
- **Logistic Regression (LR)** LR computes probability that a set of features will result in a particular class. LR will attempt to fit a line/plane between these probabilities when plotted in probabilty distribution space. Hyperparameters:
  C: [0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000] — Regularisation strength.
- **Random Forest (RF)** RF produces a collection of decorrelated trees then takes the average result from them. Much like a decision tree, just on a broader scale. For each split in the tree we take a random subset of features, which prevents over fitting to a particular training set.
  Hyperparameters:
  Max Features: ['auto', 'sqrt', 'log2', None] — Number of features to consider when looking for the best fit.
  Criterion: ['gini', 'entropy'] — Function to measure the quality of a split.

Features The feature selection process reduces the number of features required to make accurate predictions. As Chen et al, we used χ^2 feature selection, with $k = 1000$. A χ^2 score is calculated measuring the strength of association between each feature and the target; if the feature is independent of the target then it's unlikely to be a strong predictor. The χ^2 test is defined as

[5] Using `sklearn.model_selection.StratifiedKFold`, `sklearn.model_selection.GridSearchCV` and `sklearn.metrics`. Random seed is always 42.

follows:

$$\chi^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}$$

Where O is the number of observations of the type i, or observed frequency, E is the expected frequency based on a distribution [22]. With $k = 1000$ only the top 1000 scoring features would be used for the machine learning process. These movies were binarized into a sparse matrix to reduce the data size of feature set, this converts a user's list of movies into binary columns indicating whether a user has such a movie on their profile.

3.3.2 Results

The results are displayed in Table 3.2; results from the Chen et al paper are in the first row in grey. The direct point of comparison is the other row in grey, using only profiles with at least 10 movies. We note that the AUC is higher across the board, with LR being 20 percentage points (pp) higher and RF and LR 10pp higher. LR is also 7pp higher for accuracy while the others are 1pp lower. For us the best classifier is logistic regression, which differs from Chen et al, whose best classifier was Naive Bayes. The results seem reasonable with the differences likely due to changes in the original dataset or a different approach to data cleansing. The higher LR results may be a consequence of better parameter search.

Including all users gives lower accuracy, while users with ≥ 5 movies gives intermediate results — this suggests that the more features a user has the stronger the predictive power is. LD dramatically affects the number of unique features, and also reduces the number of users by roughly 25% (unmatched movies were removed). Since combining titles with LD might be incorrect and doesn't lead to empirical improvements we do not pursue it further. Overall, we can conclude that Chen et al's choice of ≥ 10 movies was appropriate and we shall continue with this criterion for the remainder of the thesis.

				Majority Baseline		NB		LR		RF	
Approach	# Users	# Features	# Unique Features	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC
Chen	?	?	?	?	?	0.837	0.824	0.774	0.712	0.827	0.785
Movies	202,578	935,662	32,024	0.509	0.502	0.726	0.818	0.729	0.821	0.712	0.797
Levenstein 0	142,982	501,453	3,134	0.509	0.500	0.754	0.832	0.754	0.833	0.732	0.804
Levenstein 1	146,413	526,500	3,238	0.513	0.505	0.752	0.831	0.752	0.832	0.731	0.801
Levenstein 2	153,830	566,668	3,369	0.508	0.500	0.747	0.827	0.747	0.829	0.727	0.800
>= 5 movies/user	69,853	673,848	26,134	0.504	0.499	0.803	0.885	0.814	0.892	0.777	0.847
>= 10 movies/user	19,532	386,880	20,480	0.537	0.504	0.827	0.897	0.844	0.913	0.810	0.871
Levenstein 0, >= 10	8,608	151,816	2,810	0.529	0.499	0.831	0.897	0.845	0.915	0.815	0.876
Levenstein 1, >= 10	9,321	166,099	2,905	0.541	0.509	0.829	0.898	0.839	0.913	0.811	0.872
Levenstein 2, >= 10	10,288	185,557	3,049	0.535	0.507	0.826	0.896	0.842	0.914	0.805	0.867

Table 3.2: Approaches to reproducing the Chen papers results, predicting a users 'private' gender using the movies they have displayed publicly on their profile.

3.4 Replication Obfuscation

Having successfully replicated the inference attack from Chen et al., we move on to reproducing their obfuscation approach.

3.4.1 Method

The same learners, data and features from 3.3 were used here, however the evaluation framework and implementation differ.

Evaluation Framework Within the cross validation, each test fold had one of the obfuscation techniques defined below applied to it.

Learners Parameters used for the learners in this section were the best performing parameters from the profile inference.

Obfuscation Techniques Chen et al. took a number of different approaches here to determine which attributes should be obfuscated, then used a policy to determine what to do with said attributes. The four techniques they used:

- χ^2. Using the feature selection technique defined in Section 3.3, this technique would compute the χ^2 value for each feature broken up by class i.e. a list of features for males ranked by χ^2 and a list of features for females ranked by χ^2.
- **Popularity**. Each feature's importance was calculated overall based on the number of individual occurrences across the whole set.
- **Majority**. Each feature's importance was calculated based on the number of individual occurrences in a class i.e. a movie that 3 people had listed would be higher up the list than a movie two people had listed.
- **Randomly**. Features for each class were selected using a random seed.

This approach to selecting attributes for obfuscation is combined with a policy: add, remove or replace.

- **Add** policy will add the opposite gender's top rated feature (calculated using the technique above) to a users features.
- **Remove** will remove the user's top rated feature for their gender calculated using the technique above.
- **Replace** will perform both the add and remove policies.

This policy drives what is done to each feature selected by the approach. We interpreted the 'ratio' of obfuscation they use to mean the ratio of a user's features to change i.e. a ratio of 0.5 will change half the user's features. We decided not to replicate the popularity policy, as it appears the Chen et al. paper to be subsumed by the majority approach. The pseudocode in Figure 3.1 describes the obfuscation strategy.

```
$R$ is the ratio of elements to change
Create a list of features (movies) called candidates $C$
Candidates is either ordered by ${\chi^2}$, popularity or random
FOR EACH USER
    $Op$ = $C$ ordered by the opposite gender to the user
    $B$ = Current users movies ordered by $C$ for their gender
    Changed = 0
    FOR EACH $B$:
        IF strategy is 'add' or strategy is 'replace':
            add the top movie from $Op$
        IF strategy is 'remove' or strategy is 'replace':
            remove this movie
        Changed += 1
        IF Changed == ceil($B * R$)
            break
```

FIGURE 3.1: The pseudocode for applying an obfuscation technique on the profile dataset to obfuscate movies reducing the accuracy of gender predictions.

3.4.2 Results

The χ^2 strategy results are shown in Figure 3.2 compared to the Chen benchmark[6]. These results are quite similar, and the patterns for each chart seem representative of Chen. The largest difference is for the remove strategy with a 15pp discrepancy for χ^2 RF. The majority strategy was less resilient for LR and RF's add and replace policies, while the remove policies across the board look the same and add/remove for NB are the same. Again the patterns in our results showing declines as the level of obfuscation increases are slightly more consistently monotonic. The random strategy was on par with Chen's results. The main conclusions are the same: that χ^2 is the best strategy for attribute selection, and replacement the best policy. We are consequently satisfied that our approach replicates the work of [4].

[6]Results with additional obfuscation combinations can be found in Appendix A

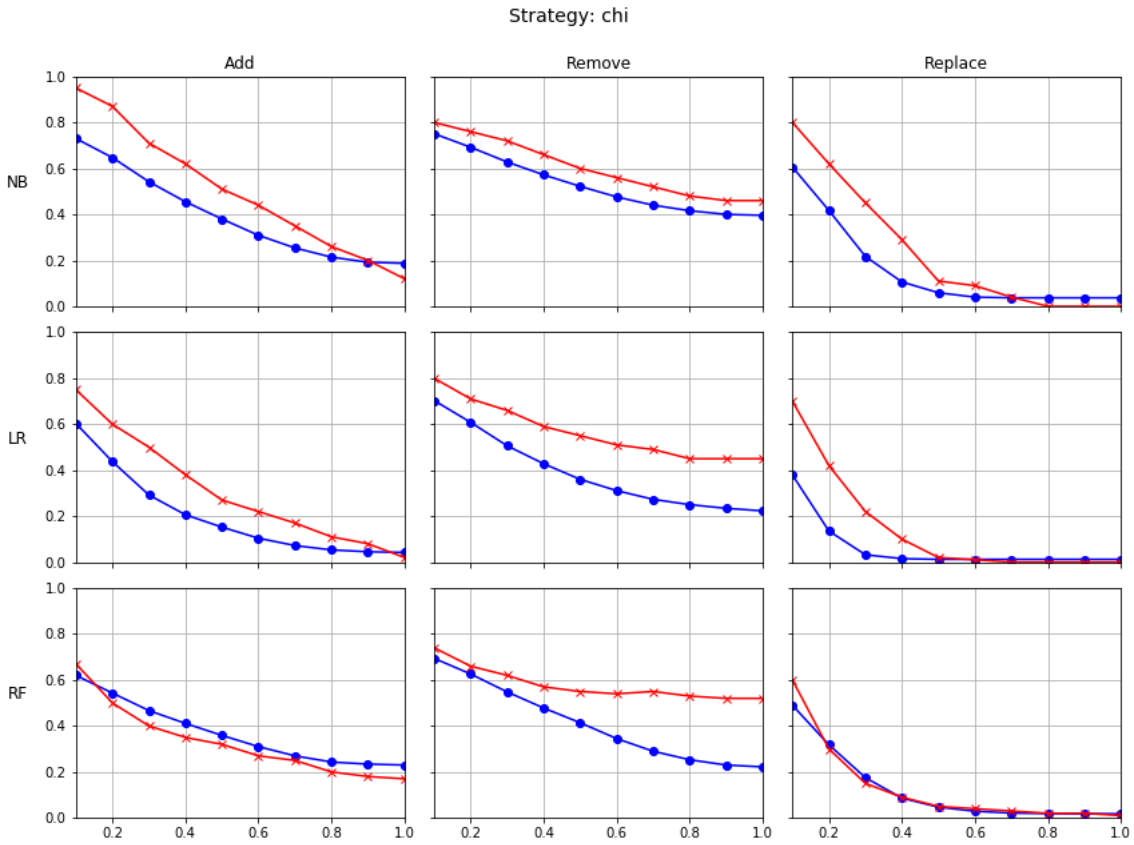

FIGURE 3.2: χ^2 Obfuscation Strategy, using movies to predict gender. Each row represents a classifier, while each columns is a different obfuscation policy. The y axis is accuracy, the x axis is the ratio of obfuscation. Blue are our results, red are Chen et al.

4

Social Network Awareness

4.1 Introduction

Your friends say a lot about you; people tend to associate with other people who are similar [11] so the characteristics that the majority of your friends have are likely characteristics you have too, a principle known as homophily. For example if you like video games it is likely that many of your friends do too. The concept of using your social network to predict attributes about you also holds for online social networks and has been applied numerous times in research with varying levels of success [5, 7, 8, 17, 19]. Gehrke et al. [24] construct an example that shows the ability to classify an individual as Democrat or Republican based simply on the proportion of their friends that have that same political affiliation. Gehrke et al. explore this example in relation to differential privacy [25], a popular privacy framework where noise defined by a distribution is added to a dataset so that privacy can be mathematically guaranteed — they show that social network information is problematic for differential privacy, motivating their proposal of a new 'zero-knowledge privacy' framework. The example from Gehrke et al. is our motivating example in what follows: we will first show that private attributes can be inferred from a user's social network properties in our Facebook data, and explore approaches to obfuscation against such inference; we will then combine these principles with our profile inference approach from Chapter 3 and the obfuscation against such an attack.

4.2 Social Network Inference

4.2.1 Data

Social network data was sourced from Data61 after they collected it as part of their work in detecting Facebook 'like farms' [26]; these are services which add artificial likes to a Facebook page, boosting its popularity and profitability. Users who *liked* specific 'honey pot' pages (which were set up for the project) had their publicly listed friends scraped, some of these pages were promoted via Facebook advertisements.

The dataset was a SQLite database `friendship.sq3` containing a specific table with mappings from a user to their friends. A Python script was used to push this information into our existing MySQL database from Chapter 3. We connected this mapping with the Chen et al. [4] profile attributes explained in Chapter 3; users who did not have any friends or did not have any friends who had specified their gender as male or female were excluded from the set. The data is summarised in Table 4.1 with the distribution of friends shown in Figure 4.1; this degree distribution exhibits a heavy tail/power law that is typical in complex networks. We verified and

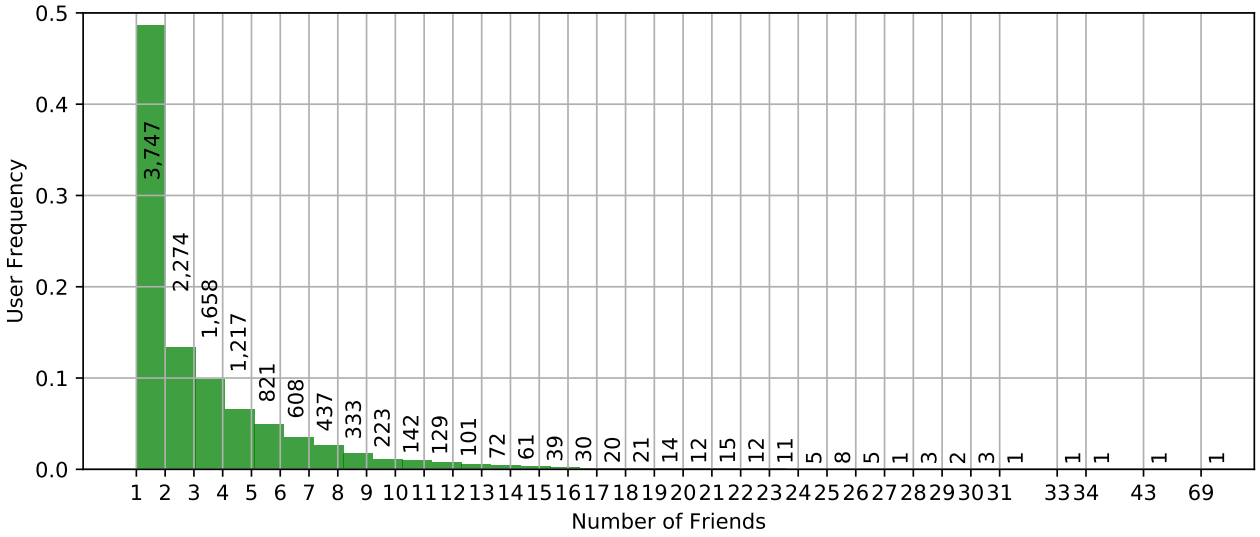

FIGURE 4.1: Network user distribution, where the x axis is the number of friends and the y axis is the number of users with that number of friends.

augmented (in a minor way) some of this public information using a selenium script,[1] which added some additional friend connections and gender/interested_in attributes.

4.2.2 Method

Evaluation Framework Stratified 10 fold cross validation (See 3.2) was used with a tuning function to determine the best parameters for the estimator. The tuning function was set to optimise for accuracy and to compute both accuracy and AUC. The mean from the cross validation results for the best estimator were used as the results for that estimator.

Learners Four classifiers were used, a majority baseline classifier as well as Logistic Regression, Naive Bayes and Random Forest. The same tuning parameters mentioned in 3.2 were used here.

Features We use just three features, motivated by [24], aggregated to the user level: count of male friends, count of female friends and ratio of male to female friends. We observe from Table 4.1 that in the aggregate there is a marked difference in the gender proportions of friends for male and female users.

4.2.3 Results

All three of the classifiers outperformed the baseline majority classifier by large margins (Table 4.2).[2] Logistic Regression had the highest accuracy (0.635) and was 0.3 percentage points from the

[1] https://www.seleniumhq.org/projects/webdriver/

[2] The same experiment was run without the 'ratio of male to females' feature — there was no notable difference to our results. We will continue using it as an attribute.

Class	Count	Friend Class	Count	%	% of Total
0 (Female)	6,288 (52.3%)	0 (Female)	14,490	65.2%	33.3%
		1 (Male)	7,749	34.8%	17.8%
1 (Male)	5,741 (47.7%)	0 (Female)	9,948	46.8%	22.9%
		1 (Male)	11,314	53.2%	26.0%
Total	**12,029**		**43,501**		

TABLE 4.1: Description of the network user data for users that have at least one friend who is male or female.

Classifier	MAJ	LR	NB	RF
Accuracy	0.509	0.635	0.603	0.626
AUC	0.508	0.666	0.638	0.669

TABLE 4.2: Results from using network information (number of male friends, number of female friends, ratio of male to female friends) to predict gender. The majority classifier is a baseline.

highest AUC, which was Random Forest on 0.669. The numbers of male and female friends are thus useful as classification features at the level of individual users, as well as being clearly different in the aggregate. These numbers are lower overall than the profile based inference results however Logistic Regression was still the best performer but with a much higher accuracy (0.844) and AUC (0.913).

4.3 Social Network Obfuscation

4.3.1 Method

The same learners, data and features from 4.2.2 were used here, however the evaluation framework and implementation differ.

Evaluation Framework Under cross validation each test fold had one of the obfuscation techniques defined below applied to it.

Learners Parameters used for the learners in this section were the best performing parameters from the network inference.

Obfuscation Techniques There were a number of potential strategies that could have been used on our dataset to obfuscate the private gender attribute. The approaches chosen are outlined below, and are broadly analogous to the profile-based approaches of Section 3.4. All take as a parameter a ratio r which dictates to what degree the obfuscation has been applied at a row level, with 0 being no obfuscation. The ratio is applied slightly differently depending on the technique. For the following we adopt the notation u for user, u^o for a users friends of the opposite gender, u^s

for users friends of the same gender. These techniques are 'aware' of the users actual gender value (`y_true`)[3].

1. ADDOPPOSITE Add friends of the opposite gender with a ratio on the number of friends the opposite gender.

   $\forall u : u^o = \lceil u^o * (1+r) \rceil$

2. REMOVESAME Remove friends with the same gender as the user, with a ratio on the number of friends of the same gender.

   $\forall u : u^s = \lfloor u^s * (1-r) \rfloor$

3. CONVERTTOOPPOSITE Reducing the number of friends of the same gender while increasing the number of friends from the opposite gender. The ratio is on the number of friends with the same gender as the user.

   $\forall u : u^s = u^s - \lceil u^s * r \rceil, u^o = u^o + \lceil u^s * r \rceil$

4.3.2 Results

Results for these techniques are shown in Figure 4.2. The two approaches that remove friends (REMOVESAME and CONVERTTOOPPOSITE) dropped the accuracy of all classifiers below the baseline with just 10% obfuscation applied, with CONVERTTOOPPOSITE being the most effective obfuscation approach overall. ADDOPPOSITE is not as effective as the other approaches, with LRs accuracy only dropping below the baseline at 55% obfuscation and RF at 100% (i.e. double the number of friends added). Naive Bayes is fairly resistant in most of the obfuscation functions, and plateaus out frequently between obfuscation rates; this is likely due to NB being less sensitive to individual changes on datasets with a small number of features. In terms of social cost, adding friends has a low social impact but at the same time has a smaller obfuscation impact compared to removal approaches which conversely have a higher social cost/impact (removing friends).

4.4 Combining Inference Models

Combining network with profile inference is motivated by other work on inference in social media data which have found that a combination approach increases accuracy [5, 9, 21]. This section will detail the implementation and results for this approach.

4.4.1 Data

The profile dataset with movies (P: 19,532 users) defined in Section 3.2 and the social network (N: 12,029 users) defined in Section 4.2.2 needed to be combined for inference to take place across both datasets. P and N both included a user identifier column which was used to determine which users existed in both datasets, all users who were not in both sets were excluded resulting

[3] 'naive' approaches that did not take the users actual gender value (`y_true`) into account performed poorly in comparison and as a result were not included.

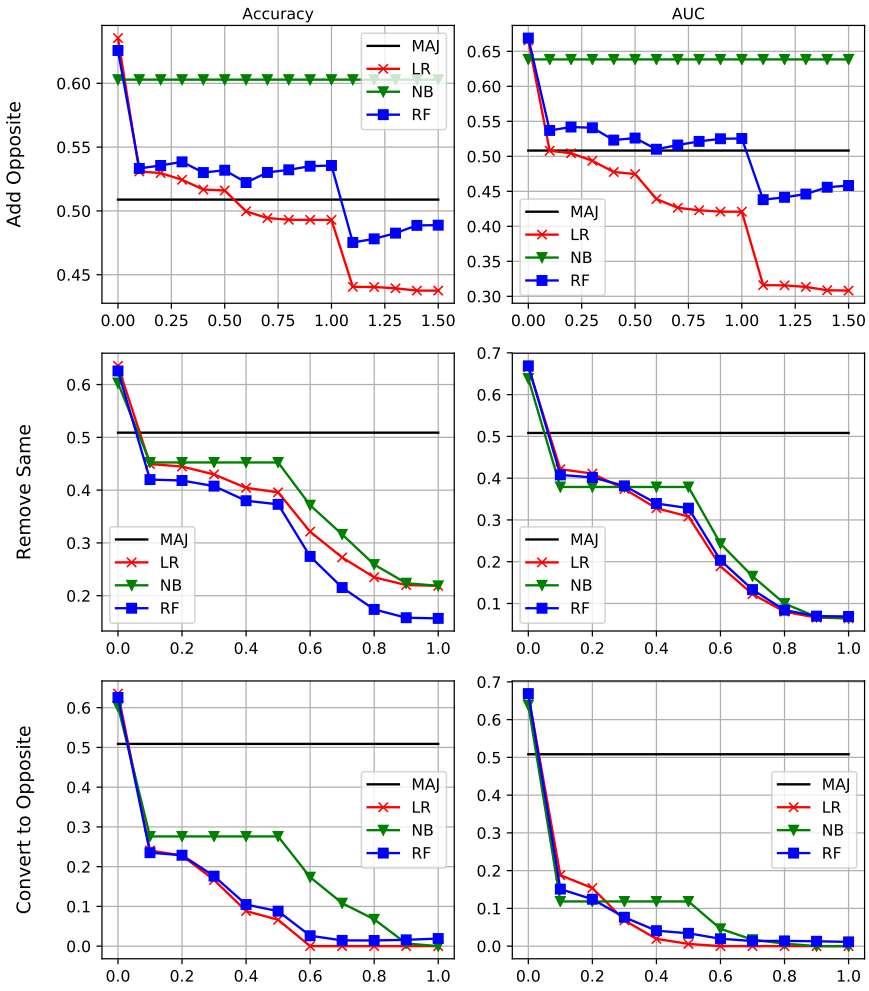

FIGURE 4.2: Obfuscation strategies applied to the network data, rows representing different obfuscation techniques, columns representing two metrics accuracy and AUC. The graph X axis is the ratio of obfuscation while the Y axis is the metrics result.

in two new datasets: P_C and N_C, both consisting of 5,177 users shown in Table 4.3. To reiterate, these users all had ≥ 10 movies on their profile, their gender was male or female and they had at least one friend with gender defined as male or female — friends not matching this description were excluded. A breakdown of these users and friends is shown in Table 4.3, noting that the proportions of genders for users and friends remains very similar to the N summary in Table 4.1.

4.4.2 Method

Evaluation Framework To augment one classifier with another classifier we decided that for each fold[4] we would take the results of the first classifier and use them as part of the feature set

[4]See cross validation Section 3.2.

Class	Count	Friend Class	Count	%	% of Total
0 (Female)	2,546 (49.2%)	0 (Female)	7049	64.7%	31.8%
		1 (Male)	3,853	35.3%	17.4%
1 (Male)	2,631 (50.8%)	0 (Female)	5,042	44.7%	22.7%
		1 (Male)	6,236	55.3%	28.1%
Total	**5,177**		**22,180**		

TABLE 4.3: Breakdown of the N_C users and friends used for the combined inference experiments.

for the second classifier[5]. We do this ensuring that we are not training on any test data in our combination system. Our process for combining the classifiers was as follows:

1. Split profile and network datasets into `trainX`, `trainY`, `testX` and `testY`. These sets should contain the same users (i.e. `trainX` for network has the same users as `trainX` for profile).
2. Train on `trainX` and `trainY` for the profile dataset
3. Run `testX` through this classifier, obtaining `predY`.
4. Add `predY` to the networks `testX`.
5. Add `trainY` to the networks `trainX`.
6. Train on the networks `trainX` and `trainY`.
7. Use the networks `testX` to obtain `predY` for network.
8. Compare the `predY` for network to the actuals (`testY`).
9. Repeat these steps for different classifiers and switching profile for network.

The results of the cross validation were averaged and presented as accuracy and AUC. The baseline metric is the classifier's performance on the dataset without using the secondary dataset and classifier (i.e. the network by itself without the profile). Note that the baseline metrics will have different values compared to those in previous chapters due to this dataset being smaller (Section 4.4.1).

The two approaches above will be defined abbreviated as follows:

- **NETPRO** The network classifier is trained first, and its results fed into the profile classifier.
- **PRONET** The profile classifier is trained first, and its results fed into the network classifier.

Learners Logistic Regression, Naive Bayes and Random Forest were used across both datasets.

Features A user's movies were used as features, with only users having at least 10 movies being included in the dataset. Three features were calculated per user in the network dataset: distinct count of male friends, distinct count of female friends and a ratio of the two.

4.4.3 Results

The results for NETPRO are shown in Table 4.4, the baseline (where P is a dash) is surpassed by every classifier in both accuracy and AUC, up to around 5 percentage points. This is to be

[5]Experiments were performed testing the addition of network features to the single profile classifier but this did not produce any interesting results.

expected due to the superior inference power of the profile dataset compared to network by itself. Worth reiterating that the baseline has changed, as step 1 and 3's results are run using the entire applicable dataset (i.e. all the network data) while these tests are done using N_C and P_C.

N	LR	LR	LR	LR	NB	NB	NB	NB	RF	RF	RF	RF
P	-	LR	NB	RF	-	LR	NB	RF	-	LR	NB	RF
ACC	.650	.650	.683	.743	.592	.592	.619	.673	.644	.639	.677	.738
AUC	.698	.831	.843	.826	.648	.868	.878	.811	.687	.824	.838	.821

TABLE 4.4: NETPRO Combination Inference, with network inference (top row) feeding into the profile inference (second row), the dashed second row is a baseline with no second classifier applied i.e. network by itself.

PRONET results are shown in Table 4.5, the accuracy results seem more driven by the profile classifier, as within LR and NB the accuracy results are almost the same with or without the network properties. RF accuracy increases from the baseline by 0.017 for all secondary classifiers, while AUC drops 0.011 when using LR or NB as the secondary classifier — using RF again as the secondary classifier reduces AUC the most (-0.052). Conversely, AUC dropped across the board compared to the baseline; the highest AUC of 0.914 from NB on profile alone. Overall, profile-based inference performance is not greatly improved by adding the simple network features, which is perhaps not surprising as there are only three of them.

P	LR	LR	LR	LR	NB	NB	NB	NB	RF	RF	RF	RF
N	-	LR	NB	RF	-	LR	NB	RF	-	LR	NB	RF
ACC	.833	.833	.833	.833	.850	.856	.856	.856	.795	.808	.814	.812
AUC	.908	.874	.868	.833	.914	.891	.887	.856	.866	.855	.856	.814

TABLE 4.5: PRONET Combination Inference, with profile inference (top row) feeding into the network inference (second row), the dashed second row is a baseline with no second classifier applied i.e. profile by itself.

4.5 Obfuscation Combination

This section will discuss the impact of obfuscation against the NETPRO and PRONET combination approaches. We will first investigate the effectiveness of the profile-based obfuscation strategies of Section 3.4 against inference classifiers using the network data. We will then apply both obfuscation approaches to see the effectiveness.

We introduce the notation (first classifier) → (second classifier) to signify the classifiers used in NETPRO and PRONET inference.

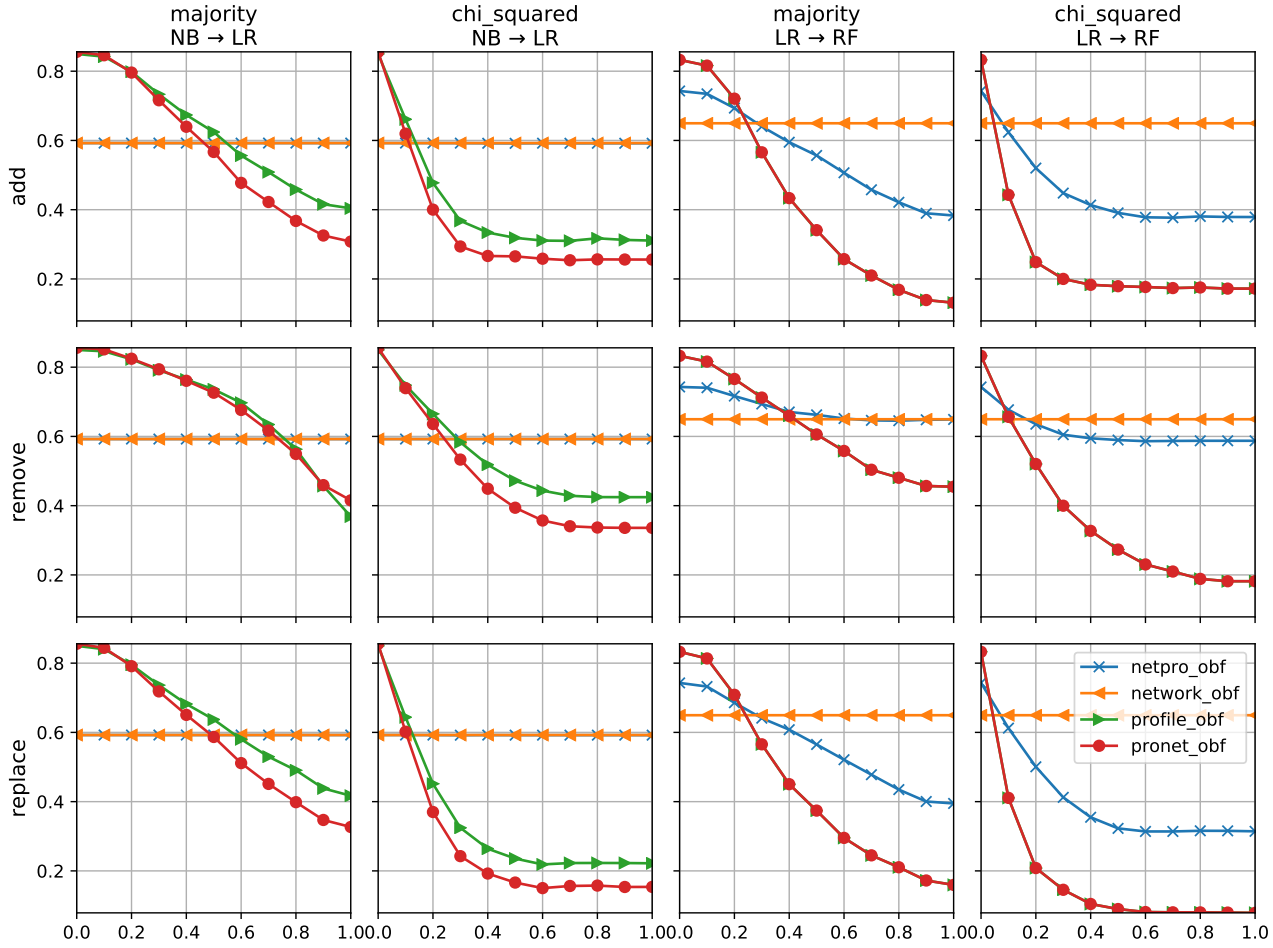

FIGURE 4.3: Profile only obfuscation approaches applied against four inference techniques predicting gender. Rows represent obfuscation policies while columns represent classifiers used and obfuscation strategy.

4.5.1 Profile Obfuscation

To examine the profile obfuscation approaches we used the best performing inference combinations from Section 4.4.3: NB → LR and LR → RF. To observe how these perform under the best obfuscation strategy (χ^2 Section 3.4.2) and also under a middling approach (majority). Results for each combination are shown in Figure 4.3.

In the LR → RF results we saw that PRONET and pure profile obfuscation will always overlap, however NETPRO is more resilient to obfuscation against the profile; showing that PRONET inference does not utilise the network features as well. Conversely in the NB → LR results we saw that NETPRO does not sway from the pure network obfuscation results.

There is less deviation in the NB → LR cases, with PRONET under performing compared to pure profile inference at $> 10\%$ obfuscation. The χ^2 approaches are still more effective at obfuscating across the board, with REPLACE again being the best policy.

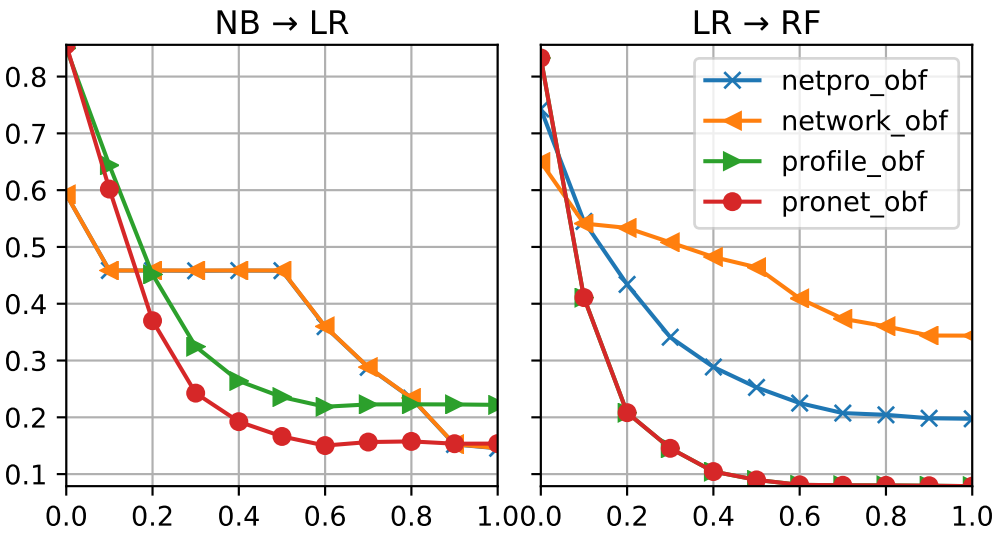

FIGURE 4.4: Obfuscating profile using χ^2 replace and network using CONVERTTOOPPOSITE. x is obfuscation ratio, y is accuracy.

4.5.2 Obfuscation of Profile and Network

The best performing profile obfuscation technique (χ^2 replace) and best performing network technique (CONVERTTOOPPOSITE) were then applied together against the combined best inference classifiers (NB → LR, LR → RF), with results shown in Figure 4.4. In both NB → LR and LR → RF PRONET is unaffected by the network obfuscation. However NETPRO drops 10-15pp in both cases but is more resilient to the obfuscation when using LR → RF classifiers. NETPRO in NB → LR plateaus after 10% obfuscation— declining again at 50% to meet PRONET at 15% accuracy (lower than profile).

4.6 Summary

With pure network inference we were able to reach an accuracy of 63.5% against a 50.9% majority class baseline; we found that the most effective obfuscation approach against network inference was CONVERTTOOPPOSITE. Profile based inference performance was not greatly improved by adding simple network features however their inclusion did make the profile obfuscation approaches less effective. χ^2 profile obfuscation approaches were still the most effective against this combined inference, in line with pure profile inference and obfuscation.

5 Imbalanced Classes

5.1 Introduction

In the previous chapters we have replicated the work of [4] in inference and obfuscation applied to datasets with roughly balanced classes (gender) and extended it to use social network structure information. This chapter is concerned with inference and obfuscation applied to imbalanced classes. Highly populated towns versus smaller towns, large employers versus start-ups and prestigious schools compared to public schools are all examples of imbalance that may impact and sway results. One attribute derivable from our data that leads to interesting imbalanced classes is sexual orientation (combining gender and interested_in). The question of how well sexual orientation can be predicted from social media or other public information is of recent scientific interest [12] and compromising such a private attribute is potentially controversial[12]. Kosinski et al. [12] are able to achieve an AUC of 88% for correctly predicting males interested in males, and 75% for females interested in females. Orientation can be seen as a sensitive attribute; correctly identifying a user as heterosexual is unlikely to cause social impact while correctly identifying a user as the inverse could have consequences. We note that accuracy is a less useful metric when classes are imbalanced (see Table 1.2), as detailed by Lopez et al [11] so F1 is used as an additional metric throughout this chapter. This case is an instance of the general observation by [11]: "The minority class usually represents the most important concept to be learned".

This chapter will explore predicting orientation from a user's movies and from network structure, examining the effects of different methods for combating class imbalance, then look at obfuscation.

5.2 Data

As per Sections 3.2 and 4.2.2 the data was already available to us and required minor reworking. We encoded our target attribute based on gender and interested_in per Table 5.1 to give us a binary predictor with the positive class being the one with the higher level of privacy sensitivity and thus the highest value to predict correctly. There were 79,349 users in the profile dataset (P) who had a gender value (male or female), an interested_in value (male or female) and at least one movie on their profile. The requirement of at least 10 movies for a profile was relaxed due to the smaller dataset size after filtering gender and interested_in. The network dataset (N) was substantially smaller with only 1,367 users having the two attributes defined and at least one friend with gender and interested_in defined. Two subset datasets were created from here, P_C and

Target	Gender	Interested In
0	0 (Female)	2 (Male)
0	1 (Male)	1 (Female)
1	0 (Female)	1 (Female)
1	1 (Male)	2 (Male)

TABLE 5.1: Target encoding for sexual orientation based on gender and interested in profile values

Target	Profile Users	Network Users	Union Users
0	67,118 (95.4%)	1,328 (97.1%)	904 (97.8%)
1	3,231 (4.6%)	39 (2.9%)	21 (2.2%)

TABLE 5.2: Counts for Sexual Orientation

N_C which only included the 925 users that existed in both datasets $P \cup N$.

As shown in Table 5.2, the percentage of the positive class ranges from 4.6% in the profile, 2.9% in the network and 2.2% in the union dataset. This is a significant class imbalance which is representative of the general population. Mercer et al. [27] performed a large scale survey interviewing 15,162 people from which we see that across the population 2.4% of males identify as gay and 1.2% identify as lesbian, with 0.7% and 2.0% of men and women respectively identifying as bisexual. Population proportions are highest for the 25-34 age group (which constitute 63% of our Facebook dataset), and are also higher for those with a college education (which Facebook users are slightly skewed towards).[3] We note as a consequence that the proportions for our sample are believable and representative.

A breakdown of the friend orientation distribution is shown in Figure 5.3. While we can see that users have very few homosexual friends overall, there is a clear proportional difference between homosexual's friends being homosexual compared to straight people with homosexual friends. The network distribution for users and friends with orientation defined is shown in Figure 5.1. Note that we have no users with more than 9 friends as a consequence of restricting ourselves to the intersection of our two datasets; the absolute number of friends is much smaller than is

Dataset	N		N_C	
User\Friend	0	1	0	1
0	1,995 (97.75%)	46 (2.25%)	1,449 (98.17%)	27 (1.83%)
1	46 (76.70%)	14 (23.30%)	26 (81.25%)	6 (18.75%)

TABLE 5.3: The count of friends in each class, rows are the base users orientation, columns are the friends orientation. Left is the entire network N, right is the network subset N_C. This is not a unique count of friends (i.e. if two users share a friend they will be counted twice).

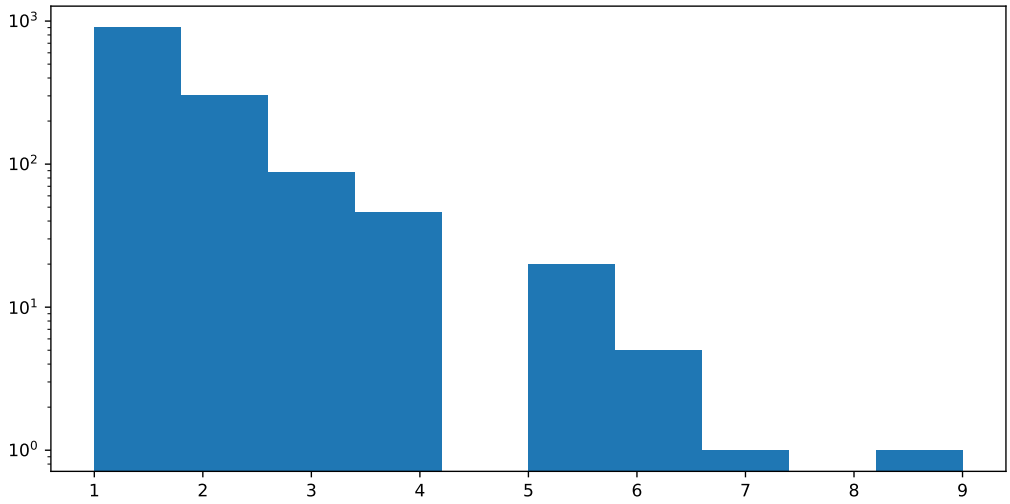

FIGURE 5.1: User to friend distribution from the social network data. The x axis is the number of friends and y axis is the number of users with that number of friends. Note that the y axis is on a log scale.

actually the case on Facebook [28].

5.3 The F1_POSITIVE Metric

It is observed in the literature [11, 29, 30, for example] that accuracy and AUC may not be the best metrics when benchmarking a classifier's effectiveness at predicting a single class, as they obscure the effect on the minority class, as illustrated by the example of [30] where the minority class in mammography-based cancer detection is the one of interest. Precision and recall are more relevant metrics; F1 as a single metric, taking the harmonic mean of precision and recall, is also appropriate:

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

We posit a scenario where an attacker would be looking to positively identify someone as attracted to the same gender, i.e. our positive class shown in Table 5.1. The F1 for the positive class is a more natural metric in this case than the macro or micro averaged F1 [4]. We will refer to this metric as F1_POSITIVE, and evaluate how inference and obfuscation apply under this metric.

[1] https://www.theguardian.com/world/2017/sep/08/ai-gay-gaydar-algorithm-facial-recognition-criticism-stanford

[2] https://www.theguardian.com/technology/2017/sep/12/artificial-intelligence-face-recognition-michal-kosinski

[3] https://sproutsocial.com/insights/new-social-media-demographics/#facebook

[4] "A macro-average will compute the metric independently for each class and then take the average (hence treating all classes equally), whereas a micro-average will aggregate the contributions of all classes to compute the average metric." https://datascience.stackexchange.com/a/24051

5.4 Method

Evaluation Framework A grid search using stratified 10 fold cross validation was performed per Section 4.2.2. Metrics were computed against four datasets: profile P, network N, profile subset P_C (i.e. only profiles that have a corresponding network entry) and network subset N_C (i.e. only network users that have a corresponding profile). The cross validation was optimised for F1_POSITIVE as opposed to the default accuracy optimisation, meaning that parameters were chosen that computed the predictions with the highest F1_POSITIVE result. For evaluating combined classifiers and dataset we used the setup of Section 4.4.2.

We evaluate both profile and network inference separately as per Sections 3.3 and 4.2.2, and then combined as per Section 4.4.

Learners The same learners and parameters were used from Section 3.2.

Features The profile dataset had the movies binarized per Section 3.2, P having 20,198 unique features and P_C having 3,692. The network dataset had three features computed: distinct count of orientation positive friends, distinct count of orientation negative friends and a ratio of positive friends to negative friends. The use of a ratio is again prompted by Gehrke et al. [24].

5.5 Results

5.5.1 Profile and Network Separately

AUC As can be seen in Table 5.4, results are generally much higher than F1_POSITIVE; even in cases where F1_POSITIVE is zero, AUC still manages a score >0.5. These AUC scores are less than Kosinski et al. [12], who had 88% in the male class and 75% in the female class, while our highest is 71%. This could be due to the differences in datasets, with theirs being a slightly different demographic or their incorporation of different features. We observe that AUC scores can conceal positive class performance: here, a high AUC can be achieved while predicting little to none of the positive class correctly.

F1_POSITIVE As can be seen in Table 5.5, NB is the highest across three of the four datasets. RF performs particularly badly on the subset datasets and the network set, with a score of zero. As our network is so small with most users only having one or two friends it makes sense that the classifiers would have a difficult time distinguishing the positive classes. P_C only has 3,692 unique features, compared to 20,198 in the full P— it is likely that a number of strong indicators were lost when reducing the set to match the smaller N. In Chapter 4 we found that profile inference was substantially stronger compared to the network; while for predicting orientation the profile is weak and in some cases provides absolutely no results at all.

Overall, while AUC and F1_POSITIVE scores cannot be directly compared, they do behave quite differently here: the AUC score is moderately good, while the F1_POSITIVE score is poor, indicating both both low precision and recall for the positive class.

Clf	P	P_C	N	N_C
LR	0.688	0.509	0.666	0.636
NB	0.707	0.562	0.618	0.560
RF	0.616	0.505	0.626	0.522

TABLE 5.4: AUC results for predicting orientation, with P and P_C using profile inference and N and N_C using network inference. These results are computed separately for each of the three classifiers.

Clf	P	P_C	N	N_C
LR	0.119	0.000	0.040	0.067
NB	0.160	0.051	0.203	0.050
RF	0.104	0.000	0.000	0.000

TABLE 5.5: F1_POSITIVE results for predicting orientation, with P and P_C using profile inference and N and N_C using network inference. These results are computed separately for each of the three classifiers.

5.5.2 Profile and Network Together

The top result for PRONET and NETPRO are shown in Table 5.6. There is a distinct difference between F1_POSITIVE, accuracy and AUC. All classifiers have high levels of accuracy and reasonably high AUC but F1_POSITIVE indicates that most combinations do not get a single prediction correct. There is only one result with a higher F1_POSITIVE than the classifiers applied to either subsets of P or N; this is the NB→NB NETPRO classifier with 0.073, 0.006 higher than the N_C result. As NB was the only classifier to achieve a result on P_C it makes sense that the highest result uses NB on the profile. Even though LR had the highest F1_POSITIVE on N_C, NB used in the combination scores higher — this is likely due to NB sacrificing accuracy to increase its F1_POSITIVE.

5.5.3 Obfuscation

The primary candidate for obfuscation was the NETPRO combination of NB classifiers as it had the highest F1_POSITIVE result (Table 5.6). Profile obfuscation was performed against pure profile inference and NETPRO inference resulting in Figure 5.2. The effectiveness of the majority remove approach is reduced when using NETPRO inference; instead, unlike for gender, majority replace and majority add are the most effective obfuscation approaches after 20% obfuscation. Network obfuscation was applied against NETPRO inference too, graphed in Figure 5.3 it shows how after 20% obfuscation the methods plateau out. These methods were shown to be very effective in Section 4.5.2 — the conclusion drawn here is that with so few friends only a small amount of obfuscation needs to take place to throw off an attacker, adding friends is completely ineffective,

1st Clf	2nd Clf	Inference	F1_POSITIVE ↓	Accuracy	AUC
NB	NB	NETPRO	0.073	0.919	0.568
NB	LR	PRONET	0.050	0.873	0.639

TABLE 5.6: The top result for each inference type when predicting sexual orientation.

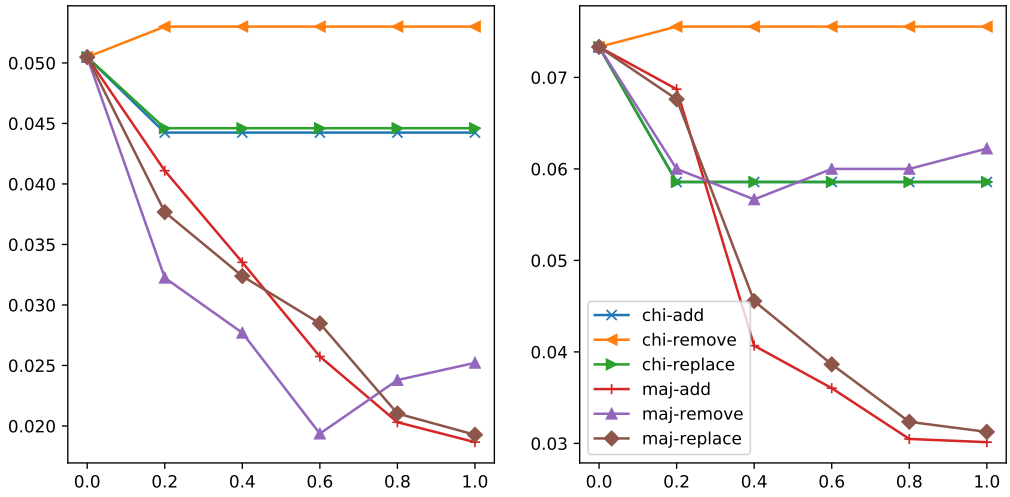

FIGURE 5.2: Profile only obfuscation results for predicting orientation. x axis is the degree of obfuscation and the y axis is the F1_POSITIVE. Left: Using profile inference on the P_C. Right: Using NETPRO inference on P_C & N_C. NB is used as the classifier(s).

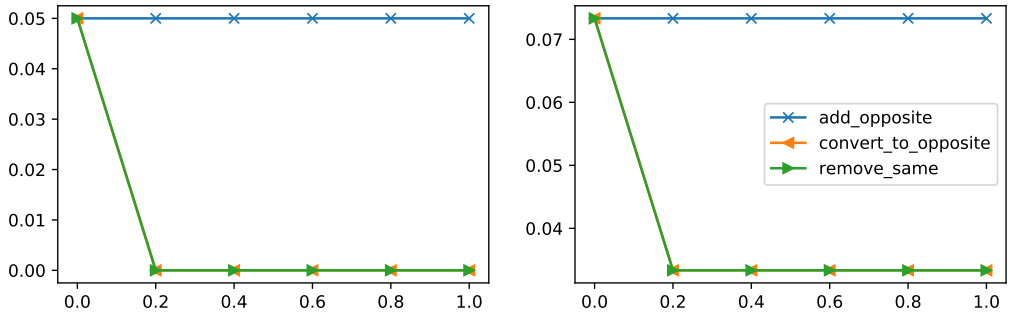

FIGURE 5.3: Network only obfuscation results for predicting orientation. x axis is the degree of obfuscation and the y axis is the F1_POSITIVE. Left: Using network inference on the N_C. Right: Using NETPRO inference on P_C & N_C. NB is used as the classifier(s).

but the removal of friends is incredibly effective.

5.6 Improving Inference Performance

5.6.1 Sampling

Justification Changes to how data is sampled can help reduce the effects of class imbalance [30]. Two overarching methods for this are:

- **Over-sampling** the minority class; those in the orientation positive class would be counted multiple times to counter the much larger proportion of the negative class.

- **Under-sampling** the majority class; those in the orientation negative class would be under counted/removed from the training set, bringing the number of samples closer to that of the positive class.

The `imbalanced-learn` [5] [31] library provides a number of techniques that fall under these methods, all of which are different approaches to selecting which elements to over/under sample. Details on each of the following are available in [31].

Over-Samplers RandomOverSampler, SMOTE and ADASYN.

Under-Samplers ClusterCentroids, RandomUnderSampler, NearMiss, InstanceHardnessThreshold, CondensedNearestNeighbour, EditedNearestNeighbours, RepeatedEditedNearestNeighbours, AllKNN, NeighbourhoodCleaningRule and OneSidedSelection.

Evaluation Framework Stratified 10 fold cross validation was performed on LR, NB and RF for each of the sampling approaches above. Evaluation was performed on P, P_C, N and N_C, with the best parameters for F1_POSITIVE chosen from previous results on these datasets. For each fold the sampling method was applied to the training set, and the test set was left untouched. F1_POSITIVE was measured for each fold and averaged to give a final metric.

Results For the sampled network inference we saw a subtle improvement in results from our baseline of 0.067 to 0.108, NB is the superior classifier here on both N and N_C, the sampling changes also managed to get non zero results from the RF classifier. The NB classifier performs the best throughout the changes to sampling approaches (under/over) and despite including more advanced approaches the standard random over/under samplers scored the same — suggesting that these approaches are not as useful on such a small dataset. The re-sampled profile results were not as radical with inference only increasing marginally from the baseline from 0.051 to 0.082 for P_C and from 0.160 to 0.167 on P when using SMOTE borderline1.

SMOTE borderline1 was the best performing technique for all of our profile (LR) and network datasets (NB) (equal first with random), so these were used for obfuscation and combined inference approaches, additional results available in Appendix A. Combined NETPRO inference resulted in an accuracy of 0.108 (same as pure network) and PRONET was significantly lower with 0.034 (0.082 baseline from pure profile inference).

5.6.2 Synthetic Network

Our conjecture from Section 5.6.1 is that the dataset is too small for accurate inference, particularly in terms of number of friends. Consequently, here we construct a synthetic network, using the actual users and their profiles but with a more realistic number of friends.

[5] https://github.com/scikit-learn-contrib/imbalanced-learn

Distribution To understand a typical friend distribution we looked to Ahn et al. [32] and Catanese et al. [33] who characterised it as a near powerlaw distribution.

$$p(x) \propto x^{-\alpha}$$

While Gjoka et al. [34] found that their dataset did not follow a strict powerlaw, they were able to establish two powerlaws within the distribution. If k is the number of friends then for $k < 300$, $\alpha_{k<300} = 1.32$, and for $300 \leq k \leq 5000$, $\alpha_{k\geq 300} = 3.38$. (Facebook has a maximum of 5,000 friends.) Later work by Ugander et al. [28] analysing the entire Facebook network at that time (unlike the sampling approaches of earlier work) argued that Facebook's friend distribution deviates from a strict powerlaw; but it acknowledges the powerlaw fit of [34] and also observes that "The distribution is nearly monotonically decreasing", which is a property of powerlaw.

As we only need a rough number of friends that is more realistic than the single-digit values in our dataset, we adopt a powerlaw distribution with $\alpha = 1.32$.

Orientation Given the network structure above, an inference must be made as to the orientation of each user and each friend if it is to fit into our existing model. As we do not know the overall statistics from broader datasets, we will make inferences from our existing data as to the orientation class for each user.

For each friend of a user, we decide if they are gay with some probability p. To estimate p, we use the population mean proportions of 0.2333 for gay users and 0.0225 for straight users (Table 5.3). This gives a binomial distribution given a particular number of friends and user orientation.

Framework A distribution was generated using `powerlaw` [6] with $\alpha = 1.32$ and $n = 70,349$ which is the number of users in P. Each element in the resulting distribution was treated as a user, with the number representing the number of friends that user has. Any values ≥ 5000 were set to 5000, shown in Figure 5.4. Each user was evaluated to decide which orientation class they would fall into, and each friend evaluated after that. The resulting dataset contained 68,298 negative classes and 2,051 positive classes ($\sim 3\%$). Each user was then randomly assigned a profile from P that matched their orientation class. We will refer to this dataset as SYNTH.

Results Pure network inference (Table 5.7[7]) on SYNTH had significantly higher results than the same approach on N, with SYNTH's F1_POSITIVE for LR and RF in the high fifties (0.57, 0.56), compared to 0.04 and 0 respectively. NB on the other hand decreased to 0.1 from 0.2.

The combination inference approach using P and SYNTH was much less interesting (Table 5.7), likely due to how strong results for pure network inference on SYNTH were relative to profile inference. None of the combination approaches were able to achieve higher results than LR on SYNTH: the higher up the ranking a classifier is, the more likely it just used the network's predictions.

[6] https://github.com/jeffalstott/powerlaw

[7] Extended results available in Appendix A.

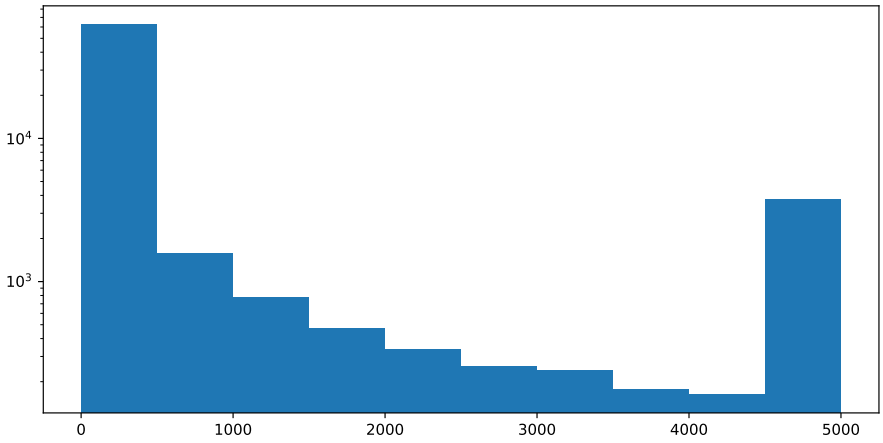

FIGURE 5.4: Synthetic network user to friend distribution, with the x axis being the number of friends and y axis being the number of users with those friends. Note that the y axis is on a log scale.

1st Clf	2nd Clf	Inference	F1_POSITIVE ↓	Accuracy	AUC
LR	-	Network-Only	0.570	0.982	0.908
LR	RF	NETPRO	0.570	0.982	0.705
LR	LR	PRONET	0.404	0.978	0.898
NB	-	Profile-Only	0.160	0.707	-

TABLE 5.7: The top result for each inference type when predicting sexual orientation using SYNTH.

Obfuscation Applying the same method as earlier, we obfuscate the combination with the highest F1_POSITIVE, here that is LR to RF on NETPRO. The results for pure profile obfuscation are completely unsurprising, as the LR approach has a score of zero for profile inference without any obfuscation there's a score for zero for any rate of obfuscation. Network obfuscation (Figure 5.5) shows the CONVERTTOOPPOSITE approach being the most effective, having dropped from 0.57 to 0.05 with 20% obfuscation. Adding is more effective than removing at lower degrees of obfuscation but loses effectiveness after 20% with a more gradual decline, while removal sees a rapid drop after 50% obfuscation. These are similar results to profile+network obfuscation on gender in Section 4.6. The results on SYNTH are the same as the results when combined with $P_{\subset}$, which suggests the networks superior inference outweighs any impact the profile could have. The NETPRO approach and the pure network approach had the exact same results.

5.7 Summary

Results using profile and network inference measured under F1_POSITIVE differ significantly from AUC, the metric used in previous work. Under this F1_POSITIVE metric, inference results are much lower than would be expected from previous work, although feature sparsity or uniqueness of our dataset could also be factors. These low inference results are easier to obfuscate against than the previous chapters high accuracy results. We implemented two approaches to boost inference, over- and under-sampling over the dataset, and creating a synthetic network with a realistic number of friends; under these, inference was much higher. Profile inference was less affected by sampling, continuing to show that profile obfuscation for our imbalanced classes was mostly pointless; on

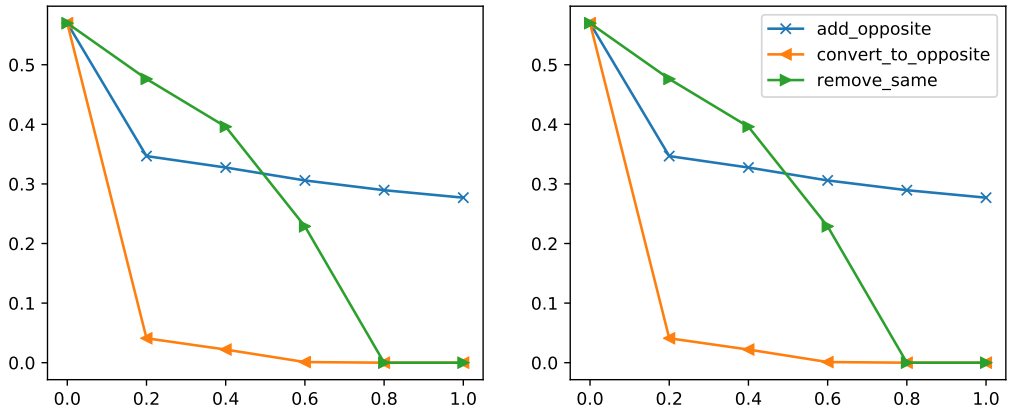

FIGURE 5.5: Network inference obfuscation results when predicting orientation. x axis is the degree of obfuscation and the y axis is the F1_POSITIVE. Left: N_C Right: NETPRO with SYNTH. Using LR→RF.

the contrary, the network obfuscation techniques had similar effectiveness as per the balanced gender classes with CONVERTTOOPPOSITE being the most effective.

"Let me give you some advice. Never forget what you are, the rest of the world will not. Wear it like armour and it can never be used to hurt you."

Tyrion Lannister

6 Conclusion

We have successfully replicated the work by Chen et al. [4] on profile inference, and shown how this can be combined with network inference and have obfuscation applied against it. We summarise below our answers to the questions posed in Chapter 2.

Q1a: How well does network-based inference work on Facebook data, and what obfuscation techniques work against it? Network inference was somewhat effective, reaching an accuracy of 63.5% against a 50.9% majority class baseline. We have explored network obfuscation in a number of different ways, showing that approaches that are aware of a user's private value are substantially more effective than naive approaches. These approaches consisted of adding friends of the opposite gender, removing friends of the same gender and a combination of the two (CONVERTTOOPPOSITE). The best performing obfuscation approach was the CONVERTTOOPPOSITE strategy.

Q1b: What's the effect of profile-based obfuscation on profile+network (P+N) inference? We have shown that P+N inference accuracy results are approx 5% higher than profile inference alone, demonstrating the claim that incorporating network structure would strengthen an attacker's inference ability, at least on a synthetic dataset that is sufficiently large enough to evaluate. Obfuscation on profile attributes alone has diminishing returns at higher levels of obfuscation when using P+N inference. P+N inference still performs better under higher levels of obfuscation than pure profile inference. In summary we show that network inference approaches can be used to counteract profile-only obfuscation mechanisms, so a combination of profile and network obfuscation must be applied.

Q1c: How can obfuscation be extended against P+N inference? We show that χ^2 REPLACE combined with CONVERTTOOPPOSITE obfuscation can decrease F1_POSITIVE by 15pp for P+N inference at just 10% obfuscation against the best PRONET combined classifier. Obfuscation against the best NETPRO classifier is shown drop roughly the same amount consistently against the previous results.

Q2a: What's the effect of profile-only inference and obfuscation under a metric reflecting class imbalance? With some discussion around metrics we determined that Accuracy and AUC were not suitable metrics while working with class imbalance under our posited sexual orientation scenario, so we used F1 of the positive class instead. Profile-only inference was not at all effective our dataset, with results being $\leq 16\%$ using a NB classifier; poor inference performance makes obfuscation easier resulting in the χ^2 techniques plateauing out in their effectiveness.

Q2b: And for P+N inference? Inference was slightly improved using our combined approaches, with majority obfuscation strategies becoming more effective than χ^2 due to them plateauing out on the profile only inference. We used two approaches to improve inference: sampling and a synthetic network. Sampling only made small improvements, while the synthetic network (constructed from our existing network distribution) improved dramatically up to 57% F1_POSITIVE.

Q2c: What is the most appropriate obfuscation technique here? The most effective obfuscation technique against the profile was MAJORITY REMOVE, which when combined with CONVERT TO OPPOSITE obfuscation on the network was able to bring the inference techniques F1_POSITIVE results down to approx. 1%.

Future Work Research in this space is beneficial to the privacy of all users online—there will be continued innovation on the inference attacker side, so it is important that there is equal development on the obfuscation side.

1. *Expand obfuscation strategies* outside of Chen et al. and simple network obfuscation, taking into account nearest neighbours and more features.
2. *More sophisticated network inference* through newer techniques like label propagation could improve our networks predictive power. Our limitations based on the dataset sparsity made this much less applicable.
3. *Include more profile features* we limited ourselves to movies to better replicate Chen et al. and to provide comparable results. Future work could include the rest of the profile classes (movies, books, music etc).
4. *Clustering of profile features* as the movies we had were fairly sparse (32K unique movies). We expected orientation inference using these movies to be more effective, so a reduction in features by something like Labelled Latent Dirichlet Allocation [35] could solve our sparsity issues and make the orientation correlated features more evident.
5. *Evaluating on different larger real datasets* would prove the effectiveness of the different obfuscation techniques against a variety of network structures and perhaps highlight a more generic algorithm that could be proposed. A larger dataset would strengthen the arguments of this research.
6. *Calculating a utility metric* to compare the quality of data before and after obfuscation. This could potentially be achieved by surveying users similar to the approach by Chen et al. [4].
7. *Dynamically calculating the privacy risk* for a user when the features (list of movies, network structure) are modified or expanded. This would simulate a real ever-changing social network and could show effectiveness of the obfuscation techniques over time.

A
Appendix

A.1 Chen Replication Obfuscation Results

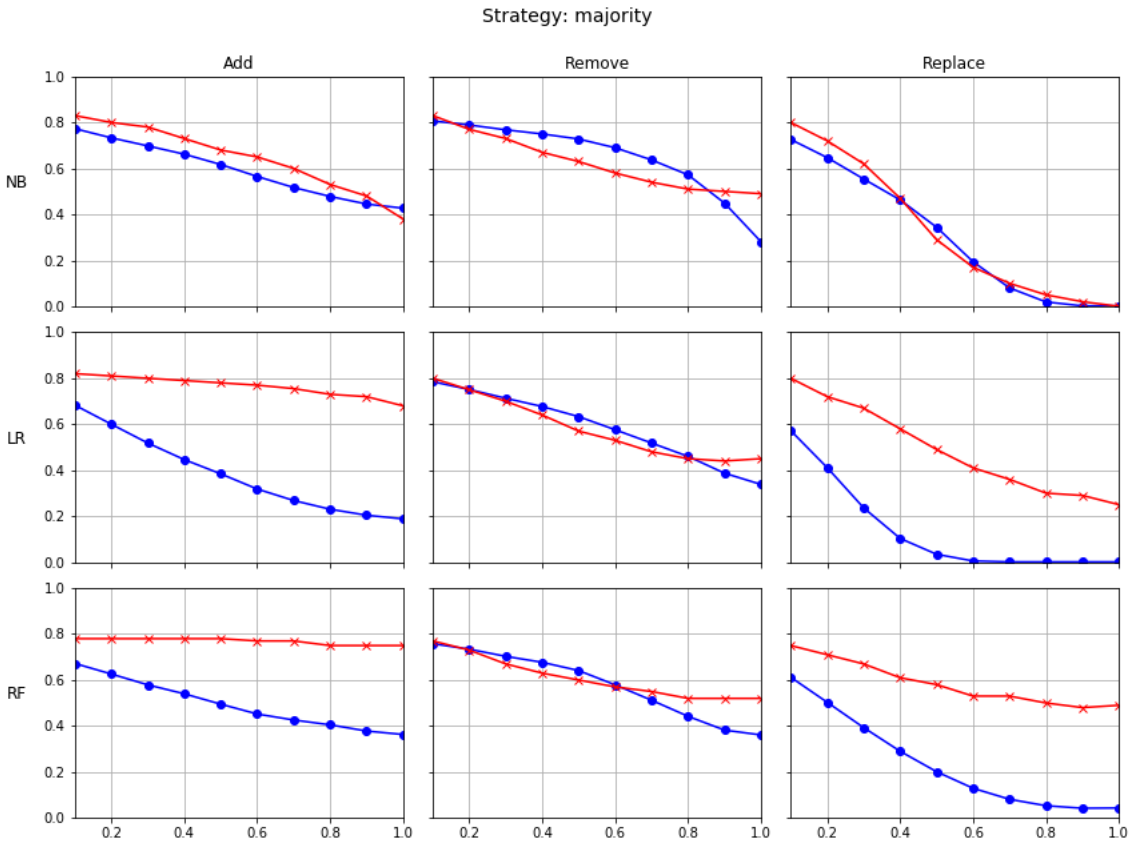

FIGURE A.1: Majority Obfuscation Strategy, using movies to predict gender. Each row represents a classifier, while each columns is a different obfuscation policy. The y axis is accuracy, the x axis is the ratio of obfuscation. Blue are our results, red are Chen et al.

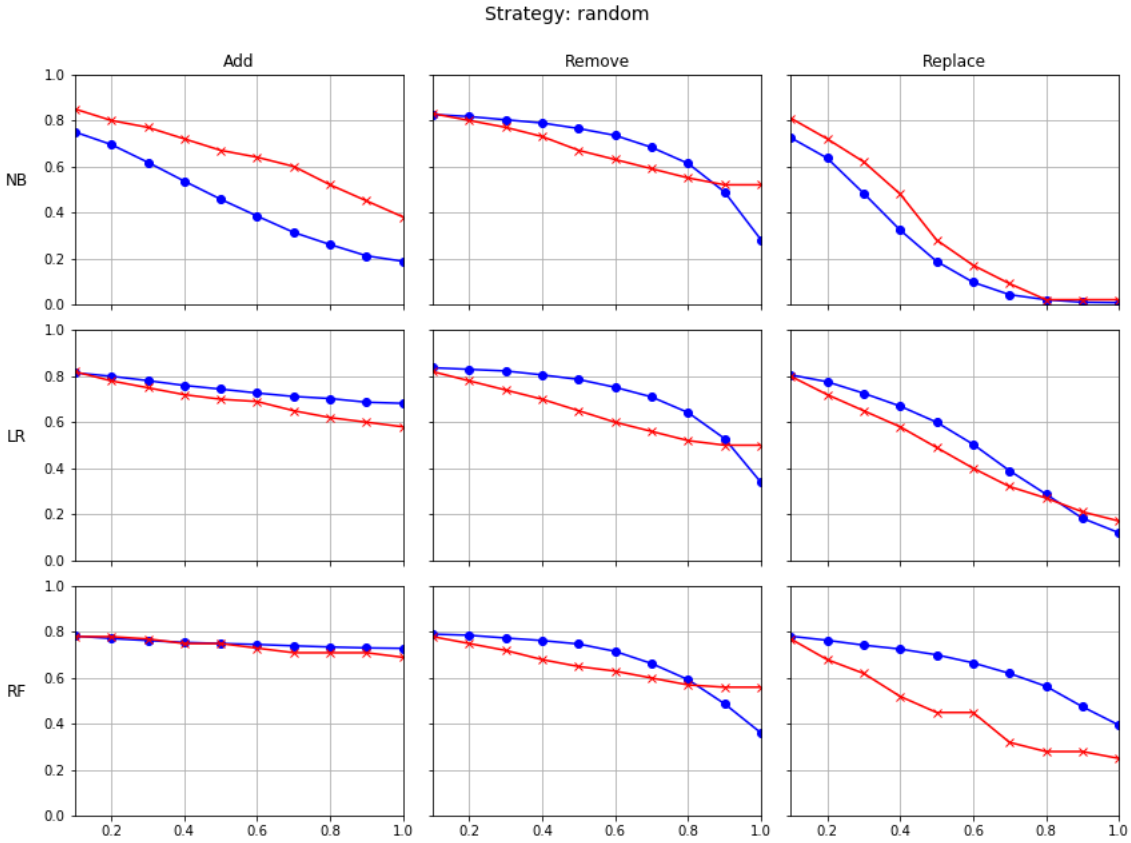

FIGURE A.2: Random Obfuscation Strategy, using movies to predict gender. Each row represents a classifier, while each columns is a different obfuscation policy. The y axis is accuracy, the x axis is the ratio of obfuscation. Blue are our results, red are Chen et al.

A.2 Sampling Results

Clf	Technique	N	$N_C \downarrow$
NB	RandomOverSampler	0.203	0.108
NB	SMOTE borderline1	0.203	0.108
NB	SMOTE borderline2	0.203	0.108
NB	SMOTE svm	0.203	0.108
NB	ADASYN	0.203	0.108
NB	SMOTE regular	0.203	0.083
LR	None	0.040	0.067
LR	SMOTE regular	0.180	0.055
LR	ADASYN	0.159	0.055
RF	SMOTE regular	0.128	0.054
RF	ADASYN	0.112	0.054
LR	SMOTE svm	0.163	0.052
NB	None	0.203	0.050
RF	RandomOverSampler	0.179	0.049
LR	RandomOverSampler	0.209	0.046
LR	SMOTE borderline1	0.041	0.034
LR	SMOTE borderline2	0.066	0.034
RF	SMOTE borderline1	0.048	0.003
RF	SMOTE borderline2	0.137	0.003
RF	SMOTE svm	0.083	0.00
RF	None	0.00	0.00

TABLE A.1: Network inference F1_POSITIVE results with over sampling methods [31] applied to the training set for predicting orientation Grey rows are the results without any sampling changes (baseline), results are ordered by their performance on N_C.

Clf	Technique	N	$N_C \downarrow$
NB	RandomUnderSampler	0.203	0.108
NB	NearMiss	0.203	0.108
LR	None	0.040	0.067
LR	NeighbourhoodCleaningRule	0.040	0.067
LR	OneSidedSelection	0.040	0.067
LR	RandomUnderSampler	0.191	0.058
NB	None	0.203	0.050
NB	NeighbourhoodCleaningRule	0.203	0.050
NB	OneSidedSelection	0.203	0.050
NB	EditedNearestNeighbours	0.170	0.050
LR	InstanceHardnessThreshold	0.058	0.050
RF	InstanceHardnessThreshold	0.053	0.050
LR	AllKNN	0.073	0.049
RF	AllKNN	0.073	0.049
LR	ClusterCentroids	0.062	0.049
RF	RandomUnderSampler	0.083	0.048
RF	ClusterCentroids	0.051	0.041
NB	ClusterCentroids	0.038	0.039
LR	RepeatedEditedNearestNeighbours	0.040	0.034
RF	RepeatedEditedNearestNeighbours	0.005	0.032
RF	NearMiss	0.055	0.030
LR	NearMiss	0.114	0.029
..	...	..	..
RF	None	0.000	0.000

TABLE A.2: Network inference F1_POSITIVE results with under sampling methods [31] applied to the training set for predicting orientation. Grey rows are the results without any sampling changes (baseline), results are ordered by their performance on N_C. Note that methods which received a zero F1_POSITIVE in either dataset were removed.

A.3 Full Orientation Results

1st Clf	2nd Clf	Inference	F1_POSITIVE ↓	Accuracy	AUC
NB	NB	NETPRO	0.073	0.919	0.568
LR	RF	NETPRO	0.067	0.978	0.510
LR	NB	NETPRO	0.059	0.927	0.564
NB	LR	PRONET	0.050	0.873	0.639
NB	NB	PRONET	0.050	0.873	0.574
NB	RF	PRONET	0.050	0.873	0.532
NB	RF	NETPRO	0.050	0.970	0.502
RF	NB	NETPRO	0.037	0.926	0.562
LR	LR	PRONET	0.000	0.977	0.615
LR	LR	NETPRO	0.000	0.977	0.514
LR	NB	PRONET	0.000	0.977	0.560
LR	RF	PRONET	0.000	0.977	0.500
NB	LR	NETPRO	0.000	0.977	0.534
RF	LR	PRONET	0.000	0.977	0.615
RF	LR	NETPRO	0.000	0.977	0.509
RF	NB	PRONET	0.000	0.977	0.560
RF	RF	PRONET	0.000	0.977	0.500
RF	RF	NETPRO	0.000	0.977	0.482

TABLE A.3: Results for predicting sexual orientation using a combination of a profiles movies and counts of their friends sexual orientation classes. Sorted by F1_POSITIVE

A.4 Full Synthetic Orientation Results

1st Clf	2nd Clf	Inference	F1_POSITIVE ↓	Accuracy	AUC
LR	-	Network-Only	0.570	0.982	0.908
LR	RF	NETPRO	0.570	0.982	0.705
LR	NB	NETPRO	0.565	0.981	0.711
RF	RF	NETPRO	0.562	0.981	0.701
RF	-	Network-Only	0.561	0.981	0.899
RF	NB	NETPRO	0.557	0.981	0.707
LR	LR	PRONET	0.404	0.978	0.898
RF	LR	PRONET	0.355	0.972	0.893
NB	LR	PRONET	0.271	0.954	0.884
NB	-	Profile-Only	0.160	0.707	-
LR	-	Profile-Only	0.119	0.509	-
NB	-	Network-Only	0.104	0.968	0.682
NB	RF	NETPRO	0.104	0.968	0.528
NB	NB	NETPRO	0.104	0.967	0.542
RF	-	Profile-Only	0.104	0.505	-
NB	NB	PRONET	0.041	0.934	0.670
NB	RF	PRONET	0.041	0.934	0.555
RF	NB	PRONET	0.017	0.961	0.677
RF	RF	PRONET	0.017	0.961	0.555
LR	LR	NETPRO	0.000	0.970	0.712
LR	NB	PRONET	0.000	0.970	0.682
LR	RF	PRONET	0.000	0.970	0.556
NB	LR	NETPRO	0.000	0.970	0.540
RF	LR	NETPRO	0.000	0.970	0.709

TABLE A.4: Results for predicting sexual orientation using a combination of a profiles movies P and counts of their friends sexual orientation classes from SYNTH. Sorted by Positive F1.

A.5 Ethics Approval

From: Faculty of Science Research Office <sci.ethics@mq.edu.au>
Sent: Thursday, June 28, 2018 3:05 PM
To: Mark Dras
Cc: fse.ethics; Katherine Shevelev; Cathi Humphrey-Hood
Subject: Ethics application 5201800389 Dras - Final Approval

Dear A/Prof Dras

RE: Ethics project entitled: "An Empirical Investigation of Privacy via Obfuscation in Social Networks"

Ref number: 5201800389.

The Faculty of Science and Engineering Human Research Ethics Sub-Committee has reviewed your application and granted final approval, effective 28/06/2018. You may now commence your research.

This research meets the requirements of the National Statement on Ethical Conduct in Human Research (2007). The National Statement is available at the following web site:

http://www.nhmrc.gov.au/_files_nhmrc/publications/attachments/e72.pdf.

The following personnel are authorised to conduct this research:

A/Prof Mark Dras
Mr Nick Reynolds

NB. STUDENTS: IT IS YOUR RESPONSIBILITY TO KEEP A COPY OF THIS APPROVAL EMAIL TO SUBMIT WITH YOUR THESIS.

Please note the following standard requirements of approval:

1. The approval of this project is conditional upon your continuing compliance with the National Statement on Ethical Conduct in Human Research (2007).

2. Approval will be for a period of five (5) years subject to the provision of annual reports.

Progress Report 1 Due: 28/06/2019
Progress Report 2 Due: 28/06/2020

Progress Report 3 Due: 28/06/2021
Progress Report 4 Due: 28/06/2022
Final Report Due: 28/06/2023

NB. If you complete the work earlier than you had planned you must submit a Final Report as soon as the work is completed. If the project has been discontinued or not commenced for any reason, you are also required to submit a Final Report for the project.

Progress reports and Final Reports are available at the following website:

http://www.research.mq.edu.au/for/researchers/how_to_obtain_ethics_approval/human_research_ethics/forms

3. If the project has run for more than five (5) years you cannot renew approval for the project. You will need to complete and submit a Final Report and submit a new application for the project. (The five year limit on renewal of approvals allows the Committee to fully re-review research in an environment where legislation, guidelines and requirements are continually changing, for example, new child protection and privacy laws).

4. All amendments to the project must be reviewed and approved by the Committee before implementation. Please complete and submit a Request for Amendment Form available at the following website:

http://www.research.mq.edu.au/for/researchers/how_to_obtain_ethics_approval/human_research_ethics/forms

5. Please notify the Committee immediately in the event of any adverse effects on participants or of any unforeseen events that affect the continued ethical acceptability of the project.

6. At all times you are responsible for the ethical conduct of your research in accordance with the guidelines established by the University. This information is available at the following websites:
http://www.mq.edu.au/policy/

http://www.research.mq.edu.au/for/researchers/how_to_obtain_ethics_approval/human_research_ethics/policy

If you will be applying for or have applied for internal or external funding for the above project it is your responsibility to provide the

Macquarie University's Research Grants Management Assistant with a copy of this email as soon as possible. Internal and External funding agencies will not be informed that you have final approval for your project and funds will not be released until the Research Grants Management Assistant has received a copy of this email.

If you need to provide a hard copy letter of Final Approval to an external organisation as evidence that you have Final Approval, please do not hesitate to contact the Ethics Secretariat at the address below.

Please retain a copy of this email as this is your official notification of final ethics approval.

Yours sincerely,
Human Research Ethics Sub-Committee
Faculty of Science and Engineering
Macquarie University
NSW 2109

References

[1] A. Narayanan and V. Shmatikov. *Robust De-anonymization of Large Sparse Datasets*. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, SP '08, pp. 111–125 (IEEE Computer Society, Washington, DC, USA, 2008). URL https://doi.org/10.1109/SP.2008.33. 1

[2] C. Culnane, B. I. P. Rubinstein, and V. Teague. *Health data in an open world*. CoRR **abs/1712.05627** (2017). 1712.05627, URL http://arxiv.org/abs/1712.05627. 1

[3] Y.-A. de Montjoye, L. Radaelli, V. K. Singh, and A. ". Pentland. *Unique in the shopping mall: On the reidentifiability of credit card metadata*. Science **347**(6221), 536 (2015). http://science.sciencemag.org/content/347/6221/536.full.pdf, URL http://science.sciencemag.org/content/347/6221/536. 1

[4] T. Chen, R. Boreli, M.-A. Kaafar, and A. Friedman. *On the Effectiveness of Obfuscation Techniques in Online Social Networks*. In E. De Cristofaro and S. J. Murdoch, eds., *Privacy Enhancing Technologies: 14th International Symposium, PETS 2014, Amsterdam, The Netherlands, July 16-18, 2014. Proceedings*, pp. 42–62 (Springer International Publishing, Cham, 2014). URL https://doi.org/10.1007/978-3-319-08506-7_3. 1, 3, 4, 6, 11, 12, 13, 19, 21, 30, 40, 41

[5] J. He and W. W. Chu. *Protecting Private Information in Online Social Networks*. In H. Chen and C. C. Yang, eds., *Intelligence and Security Informatics: Techniques and Applications*, pp. 249–273 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2008). URL https://doi.org/10.1007/978-3-540-69209-6_14. 1, 7, 11, 12, 21, 24

[6] S. Volkova, G. Coppersmith, and B. V. Durme. *Inferring user political preferences from streaming communications*. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pp. 186–196 (2014). URL http://aclweb.org/anthology/P/P14/P14-1018.pdf. 8

[7] A. Rahimi, D. Vu, T. Cohn, and T. Baldwin. *Exploiting Text and Network Context for Geolocation of Social Media Users*. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1362–1367 (Association for Computational Linguistics, Denver, Colorado, 2015). URL http://www.aclweb.org/anthology/N15-1153. 8, 9, 21

[8] A. Ramezani, M. Dras, and D. Q. Nguyen. *The Usefulness of Indirect Social Information and Generalised Content in Identifying Negative Attitudes about Vaccines on Twitter*. Mres thesis, Macquarie University (2016). 8, 9, 10, 21

[9] N. Z. Gong and B. Liu. *You are Who You Know and How You Behave: Attribute Inference Attacks via Users' Social Friends and Behaviors*. CoRR **abs/1606.05893** (2016). URL http://arxiv.org/abs/1606.05893. 10, 24

[10] D. Rao, D. Yarowsky, A. Shreevats, and M. Gupta. *Classifying latent user attributes in twitter*. In *Proceedings of the 2Nd International Workshop on Search and Mining User-generated Contents*,

SMUC '10, pp. 37–44 (ACM, New York, NY, USA, 2010). URL http://doi.acm.org/10.1145/1871985.1871993. 1, 5, 7

[11] V. López, A. Fernández, S. García, V. Palade, and F. Herrera. *An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics*. Information Sciences **250**, 113 (2013). URL http://www.sciencedirect.com/science/article/pii/S0020025513005124. 3, 21, 30, 32

[12] M. Kosinski, D. Stillwell, and T. Graepel. *Private traits and attributes are predictable from digital records of human behavior*. Proceedings of the National Academy of Sciences **110**(15), 5802 (2013). http://www.pnas.org/content/110/15/5802.full.pdf, URL http://www.pnas.org/content/110/15/5802. 3, 6, 30, 33

[13] E. Sarigöl, D. García, and F. Schweitzer. *Online privacy as a collective phenomenon*. CoRR **abs/1409.6197** (2014). 1409.6197, URL http://arxiv.org/abs/1409.6197. 3, 8

[14] N. Garera and D. Yarowsky. *Modeling latent biographic attributes in conversational genres*. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pp. 710–718 (Association for Computational Linguistics, Stroudsburg, PA, USA, 2009). URL http://dl.acm.org/citation.cfm?id=1690219.1690245. 5

[15] A. Korolova. *Privacy violations using microtargeted ads: A case study*. In *2010 IEEE International Conference on Data Mining Workshops*, pp. 474–482 (2010). 6

[16] I. Faizullabhoy and A. Korolova. *Facebook's advertising platform: New attack vectors and the need for interventions* (2018). 6

[17] D. PreoÅčiuc-Pietro, Y. Liu, D. Hopkins, and L. Ungar. *Beyond Binary Labels: Political Ideology Prediction of Twitter Users*. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 729–740 (Association for Computational Linguistics, Vancouver, Canada, 2017). URL http://aclweb.org/anthology/P17-1068. 6, 7, 21

[18] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. *Distributed representations of words and phrases and their compositionality*. CoRR **abs/1310.4546** (2013). 1310.4546, URL http://arxiv.org/abs/1310.4546. 6

[19] X. Zhou, E. W. Coiera, G. Tsafnat, D. Arachi, M.-S. Ong, and A. G. Dunn. *Using social connection information to improve opinion mining: Identifying negative sentiment about hpv vaccines on twitter*. Studies in health technology and informatics **216**, 761 (2015). 9, 10, 21

[20] Forbes. *Has Google+ Really Died?* https://www.forbes.com/sites/stevedenning/2015/04/23/has-google-really-died/#d722645466c0 (2015). URL https://www.forbes.com/sites/stevedenning/2015/04/23/has-google-really-died/#d722645466c0. 10

[21] G. B, H. T, and P. AV. *Modeling contagion through social networks to explain and predict gunshot violence in chicago, 2006 to 2014*. JAMA Internal Medicine **177**(3), 326 (2017). URL http://dx.doi.org/10.1001/jamainternmed.2016.8245. 10, 24

[22] E. Alpaydin. *Introduction to Machine Learning* (The MIT Press, 2010), 2nd ed. 15, 16

[23] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning: data mining, inference and prediction* (Springer, 2009), 2 ed. URL http://www-stat.stanford.edu/~tibs/ElemStatLearn/. 15

[24] J. Gehrke, E. Lui, and R. Pass. *Towards privacy for social networks: A zero-knowledge based definition of privacy*. In *Proceedings of the 8th Conference on Theory of Cryptography*, TCC'11, pp. 432–449 (Springer-Verlag, Berlin, Heidelberg, 2011). URL http://dl.acm.org/citation.cfm?id=1987260.1987294. 21, 22, 33

[25] C. Dwork. *Differential privacy*. In M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, eds., *Automata, Languages and Programming*, pp. 1–12 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2006). 21

[26] M. Ikram, L. Onwuzurike, S. Farooqi, E. D. Cristofaro, A. Friedman, G. Jourjon, M. A. Kâafar, and M. Z. Shafiq. *Measuring, characterizing, and detecting facebook like farms*. CoRR **abs/1707.00190** (2017). 1707.00190, URL http://arxiv.org/abs/1707.00190. 21

[27] C. H. Mercer, C. Tanton, P. Prah, B. Erens, P. Sonnenberg, S. Clifton, W. Macdowall, R. A. Lewis, N. M. Field, J. Datta, A. J. Copas, A. Phelps, K. Wellings, and A. M. Johnson. *Changes in sexual attitudes and lifestyles in britain through the life course and over time: findings from the national surveys of sexual attitudes and lifestyles (natsal)*. In *The Lancet* (2013). 31

[28] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow. *The anatomy of the facebook social graph*. CoRR **abs/1111.4503** (2011). 1111.4503, URL http://arxiv.org/abs/1111.4503. 32, 37

[29] J. Davis and M. Goadrich. *The relationship between precision-recall and roc curves*. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pp. 233–240 (ACM, New York, NY, USA, 2006). URL http://doi.acm.org/10.1145/1143844.1143874. 32

[30] N. V. Chawla. *Data Mining for Imbalanced Datasets: An Overview*, pp. 853–867 (Springer US, Boston, MA, 2005). URL https://doi.org/10.1007/0-387-25465-X_40. 32, 35

[31] G. Lemaître, F. Nogueira, and C. K. Aridas. *Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning*. Journal of Machine Learning Research **18**(17), 1 (2017). URL http://jmlr.org/papers/v18/16-365. 36, 45, 46

[32] Y.-Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong. *Analysis of topological characteristics of huge online social networking services*. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pp. 835–844 (ACM, New York, NY, USA, 2007). URL http://doi.acm.org/10.1145/1242572.1242685. 37

[33] S. Catanese, P. De Meo, E. Ferrara, G. Fiumara, and A. Provetti. *Extraction and Analysis of Facebook Friendship Relations*, pp. 291–324 (Springer London, London, 2012). URL https://doi.org/10.1007/978-1-4471-4054-2_12. 37

[34] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou. *Walking in facebook: A case study of unbiased sampling of osns*. In *2010 Proceedings IEEE INFOCOM*, pp. 1–9 (2010). 37

[35] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning. *Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora*. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pp. 248–256 (Association for Computational Linguistics, Stroudsburg, PA, USA, 2009). URL http://dl.acm.org/citation.cfm?id=1699510.1699543. 41