ABSTRACT

GARG, VAIBHAV. Using Natural Language Processing to Understand Harassment-related Posts. (Under the direction of Chair first and last name).

Abstract text ...

Using Natural Language Processing to Understand Harassment-related Posts

by
Vaibhav Garg

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

program name

Raleigh, North Carolina
2023

APPROVED BY:

Member 1 name	Member 2 name
Member 3 name	Member 4 name

Chair first and last name
Chair of Advisory Committee

BIOGRAPHY

The author was born in a small town …

TABLE OF CONTENTS

List of Tables . . . . . **v**

List of Figures . . . . . **vi**

Chapter 1 INTRODUCTION **1**
1.1 Background and Research Questions 3
1.1.1 Online Harassment 3
1.1.2 Real-World Harassment 3
1.1.3 Online Harassment Leading to Real-World Harassment 4
1.1.4 Proposal: Understanding Violation of Consent 4
1.2 Contributions and Novelty 5
1.2.1 iROGUE: Identifying Rogue Behavior from App Reviews 5
1.2.2 METHREE: Highlighting Incidents, Effects, and Requested Advice from MeToo Posts 6
1.2.3 Understanding Inciting Speech in Social Media 6
1.2.4 Proposal: Understanding Violation of Consent 7

Chapter 2 iROGUE: Identifying Rogue Behavior from App Reviews **8**
2.1 Introduction 9
2.2 App Reviews Reveal Rogue Behavior 12
2.2.1 Seed Dataset 12
2.2.2 Investigating Reviews 12
2.3 The iROGUE Approach 15
2.3.1 Computing Alarmingness of Reviews 15
2.3.2 Identifying Rogue Apps 21
2.3.3 Identifying Additional Rogue Apps 26
2.4 Uncovering Rogue Functionalities 28
2.5 Related Work 30
2.5.1 Spying through Mobile Apps 30
2.5.2 User Privacy 31
2.5.3 Using Natural Language Processing 32
2.6 Discussion 33
2.6.1 Threats to Validity 33
2.6.2 Limitations and Future Directions 34

Chapter 3 METHREE: Identifying Incidents, Effects, and Requested Advice from MeToo Posts **38**
3.1 Introduction 39
3.1.1 Research Questions 41
3.1.2 Contributions and Novelty 41
3.1.3 Key Findings 42
3.2 The METHREE Dataset and Classifier 42
3.2.1 Initial Training Data for Active Learning 43

3.2.2 Initial Model to identify Sentences . . . 47
3.2.3 Completing Active Learning Cycles . . . 49
3.2.4 Selecting Query Method . . . 50
3.3 Qualitative Analysis . . . 53
3.4 Related Work . . . 53
3.5 Discussion . . . 54
3.5.1 Conclusion . . . 54
3.5.2 Limitations and Future Work . . . 55
3.6 Broader Perspective and Ethical Considerations . . . 55

Chapter 4 Understanding Inciting Speech in Social Media . . . 57
4.1 Introduction . . . 58
4.2 Rhetorical Strategies in Incitement . . . 59
4.3 Analyzing Incitement Strategies . . . 60
4.3.1 Generality of the Strategies . . . 61
4.3.2 Textual Signatures of Incitement Strategies . . . 62
4.4 Impoliteness Super-Strategies in Incitement . . . 64
4.5 Method . . . 65
4.5.1 Curation of INCITE . . . 66
4.5.2 Model Training . . . 68
4.6 Contributions and Novelty . . . 70
4.7 Related Work . . . 70
4.7.1 Hate Speech . . . 71
4.7.2 Dangerous Speech . . . 71
4.7.3 Fear Speech and Islamophobia . . . 71
4.8 Limitations and Future Work . . . 72

Chapter 5 Understanding Violation of Consent . . . 73

References . . . 75

APPENDICES . . . 85

LIST OF TABLES

Table 1.1 Nine criteria for consent formulated by Singh (2022). 5
Table 1.2 Proposed plan. 7

Table 2.1 Details of our seed dataset. 12
Table 2.2 Occurrence of each keyword. 13
Table 2.3 Count of stories for each reviewer type. 15
Table 2.4 Performance of three regression models on ten-fold cross validation. . . . 20
Table 2.5 Score buckets for alarmingness. 21
Table 2.6 Instructions followed for labeling rogue apps. 24
Table 2.7 Choosing an appropriate threshold according to the recall scores. 25
Table 2.8 Performance (in %) of baseline methods and iROGUE on the seed dataset. Bold value for a metric indicates the highest score among all approaches. 26
Table 2.9 Apps and reviews in the snowball dataset. 27
Table 2.10 Performance (in %) of baseline methods and iROGUE on the snowball dataset. Bold value for a metric indicates the highest score among all approaches. 28
Table 2.11 Types of rogue functionalities. 30

Table 3.1 Relevant examples according to labeling instructions. 47
Table 3.2 Cohen's kappa scores for each pair of annotators. 48
Table 3.3 Comparing performance of multiple trained models. Bold value for a metric indicates the highest score among all approaches. 49

Table 4.1 Culpeper's super-strategies for impoliteness and their prevalence in a sample of 100 Islamophobic sentences. 65
Table 4.2 AUC-ROC score for each of the five folds. Bold indicates the highest average AUC-ROC score among all approaches. 70

LIST OF FIGURES

Figure 2.1 Overview of iROGUE approach. 16

Figure 3.1 Active learning involves four iterative steps. 43
Figure 3.2 Venn diagram showing the distribution of sentences across the three categories. 50
Figure 3.3 ROC curve showing true positive and false positive rates, while considering misclassified incident sentences under positive class. The area under the curve is 0.84. The best threshold is 0.038177. 51
Figure 3.4 ROC curve showing true positive and false positive rates, while considering misclassified effects sentences under positive class. The area under the curve is 0.83. The best threshold is 0.008476. 51
Figure 3.5 ROC curve showing true positive and false positive rates, while considering misclassified requested-advice sentences under positive class. The area under the curve is 0.98. The best threshold is 0.007874. 51

Figure 4.1 Three kinds of malice on social media. Hate speech involves an attacker expressing hatred toward a target. Provocative speech involves an attacker provoking a target to elicit a reaction from them (such a reaction may have the target lose face or otherwise advance the attacker's agenda). Inciting speech involves the attacker riling up an intermediary to use as an instrument in expressing hatred or carrying out a malicious action on the target. We focus on inciting speech. 58
Figure 4.2 Showing the average sentence vector for each rhetorical category after t-SNE dimensionality reduction. 62
Figure 4.3 Prominent words in the three rhetorical strategies. 63
Figure 4.4 Distribution of sentences across the three rhetorical strategies and None. 64
Figure 4.5 Overview of our method. First, we curated a dataset of inciting sentences called INCITE. Second, we leveraged INCITE to train multiple models and choose the best one. 66

CHAPTER

1

INTRODUCTION

Harassment is a well-known problem in our society. According to the Center for Disease Control and Prevention (CDC), in the United States, one in five women has experienced attempted rape, and one in four men has experienced sexual violence. With the advent of the internet, the problem of harassment is not only limited to real-world scenario (e.g., verbal, physical, or sexual harassment) but also applies to online scenario. In totality, we divide the harassment scenarios into three types: (i) online harassment, (ii) real-world harassment, and (iii) online harassment leading to real-world harassment. *Online harassment* includes targeting individuals through disturbing text or by disseminating their personal information, in turn causing them threat, harm, and distress[1]. On the other hand, *Real-world harassment* is an unwelcome conduct that is based on attributes such as sex, race, color, and so on[2]. In the third scenario, harassment is initiated in the online setting but may also spread to a real-world setting. In this thesis, we identify specific problems in each of these scenarios and computationally solve them.

In the online harassment scenario, we identify that the abusers misuse mobile apps to access information (such as location and call logs) of other users or bystanders (victims). Such apps are dual-use and violate victims' privacy expectations. App users should be aware of such apps and their functionalities before installing them. Moreover, app developers and platforms

[1] `https://privacyrights.org/consumer-guides/online-harassment-cyberstalking`
[2] `https://www.eeoc.gov/harassment`

such as Apple App Store should take measures to ensure victims' privacy. We found that both abusers and victims report such harassment in reviews of these apps. In reviews, abusers brag about using these apps to spy on or stalk their family or friends, whereas victims share their concerns and grievances, including frustration at the loss of privacy. We leverage the rich information in reviews and apply NLP techniques to identify these apps and their problematic functionalities.

For the real-world harassment scenario, we focus on the prevailing problem of sexual harassment. Survivors of sexual harassment may not be open about sharing the harassment experiences with family or friends. To seek supportive responses or advice, they leverage online platforms to anonymously post about their experiences. One such platform is Reddit that hosts multiple subreddits such as r/meToo. We found that while sharing harassment experiences on these subreddits, the survivors end up writing long posts (with mean and maximum of 1,881 and 33,432 characters, respectively). For a prospective helper on subreddits, reading long posts can be demanding and time consuming. Moreover, while framing response to the survivors, the helper may miss out on important parts of the long posts that should be addressed. We leverage these long MeToo posts (shared on subreddits) to identify three types of key information: (i) incident, (ii) effects on the survivor, and (iii) advice sought. We develop a natural language based model that can highlight this key information (in long posts) for helpers to address in their responses.

In the third scenario, we focus on the problem of inciting speech prevalent on social media. Inciting speech is the speech that can make the readers angry or urge them to act against the target. As compared to hate speech, inciting speech is subtle and hence harder to identify. On one hand, inciting speech on social media can disturb, embarrass, or threaten the target (online harassment), whereas on the other hand, it can also lead to violence against the target (real-world harassment). For concreteness, we focus on WhatsApp posts (from Indian groups discussing politics) that include inciting speech against Muslims. We found that incitement can of three forms: criticizing the target community's members and their beliefs, criticizing the target community for its imputed misdeeds, and exhortation against the target community. We leverage NLP techniques to identify inciting sentences in a social media post. In addition, we study inciting speech through a pragmatic lens by: (i) analyzing the impoliteness super-strategies and (ii) identify dialog acts relevant to incitement.

In both online and real-world scenarios, consent is an important and misunderstood subject. In many situations, the consenter misunderstands what they consented to, or incapable of giving it, or is sometimes coerced to do so. Through online and real-world harassment posts, we propose to identify and analyze how consent is violated. This will advance the research in understanding consent through natural language described in such harassment posts.

1.1 Background and Research Questions

1.1.1 Online Harassment

Chatterjee et al. (2018) use app descriptions to identify intimate partner surveillance (IPS) apps, the apps that someone can use to spy on their intimate partners (spouse, boyfriend, or girlfriend). Such IPS apps are dual-use apps, that have a legitimate purpose but are misused for spying. Their work is specific to IPS, whereas we focus on the general misuse of mobile apps for information access.

The privacy expectations of an app user or a third party (i.e., a *victim*) are violated when the victim (1) is not aware of the information access (spying) or (2) may know about the access, but is uncomfortable with it. The latter case includes incidents of forced consent or when public information on apps (such as dating platforms) is accessed beyond the victim's level of comfort (stalking). We use the term *rogue behavior* to mean these two types of information access and use *rogue apps* to mean the apps that enable rogue behavior. We use the term *rogue behavior* to mean these two types of information access and use *rogue apps* to mean the apps that enable rogue behavior.

We found that app reviews contain rich information about such misuse. As a result, we propose the following research questions:

RQ$_{\text{rogue-apps}}$. How can we identify rogue apps from reviews?

RQ$_{\text{rogue-functionality}}$. How can we uncover rogue functionalities?

Identifying rogue apps and their functionalities can help in warning users (potential victims) about potential misuse and ensuring their privacy. Moreover, app platforms and developers should take measures to meet users' privacy expectations.

1.1.2 Real-World Harassment

For understanding experiences of sexual harassment, prior studies on MeToo posts (Karlekar and Bansal 2018; Hassan et al. 2020; Khatua et al. 2018; Ghosh Chowdhury et al. 2019a) focus on classification. The expectation is that a prospective helper (on online platforms) can provide support to the survivors of the identified posts. However, merely identifying relevant posts is not enough. Prior research shows that important parts of the posts such as (i) the Incident, (ii) the effects on the survivor (Field-Springer et al. 2021), and (iii) the advice that the survivor is seeking (Andalibi et al. 2016), must be addressed in order to provide help. We propose the following research question.

$RQ_{metoo\text{-}identify}$: How can we identify sentences describing the harassment incident, its effects on the survivor, and the requested advice from a MeToo post?

In the case of long MeToo posts (especially present on subreddits), automatically identifying and highlighting the above sentences can assist helpers to address important concerns in their responses.

1.1.3 Online Harassment Leading to Real-World Harassment

Inciting speech is to be differentiated from hate speech, which is another kind of antisocial communication (Calvert 1997). Unlike hate speech (ElSherief et al. 2018), inciting speech doesn't necessarily contain derogatory language (words such as *n*gger* and *a**hole*) to attack an individual or a group. Whereas many social media studies focus on hate speech identification (Aluru et al. 2021; Das et al. 2022), identification of inciting speech has not garnered as much attention.

Our qualitative analysis revealed three rhetorical strategies used for incitement: (i) identity (attacking target members and their beliefs), (ii) imputed misdeeds (highlighting misdeeds of the target), and (iii) exhortation (calling for action). This problem is even more prominent on platforms such as WhatsApp where incitement is used against religious groups but there is no content moderation due to end-to-end encryption (Saha et al. 2021). We propose the following research questions.

$RQ_{incite\text{-}identify}$: How can we automatically identify sentences showing rhetorical strategies in a post?

$RQ_{incite\text{-}pragmatics}$: What dialogue acts and impoliteness super-strategies are used in inciting speech?

A social media post can contain multiple inciting sentences. To pinpoint each sentence along with its rhetorical strategy, sentence-level identification ($RQ_{incite\text{-}identify}$) is helpful. Moreover, to understand inciting speech through the pragmatic lens, we identify dialog acts and impoliteness super-strategies leveraged ($RQ_{incite\text{-}pragmatics}$).

1.1.4 Proposal: Understanding Violation of Consent

Consent is sometimes considered as the consenter's internal mental action (Singh 2022), but sometimes reflects a performative or communicative act (Hohfeld 1923; Koch 2018). Singh (2022) leveraged Habermas's validity claims (Habermas 1984) and proposed nine important

criteria for formulating consent. Table 1.1 summarizes those criteria and their implied explanations.

Table 1.1: Nine criteria for consent formulated by Singh (2022).

Criterion	Meaning
Visibility	Consent is observable, e.g., communication
Free will	Consenter acts without being coerced
Truth	Consenter's beliefs are true and complete
Capacity	Consenter is mentally fit
Cognition	Consenter believes and intends to
Attention	Consenter exercises mental faculties
Statutes	Consenter meets statutory criteria, e.g., age
Power	Consenter is not subjugated by consentee
Honesty	Consentee does not mislead consenter

We plan to focus on these nine criteria of consent and analyze their violations in both online and real-world harassment scenarios. We will leverage our curated datasets and explore how the violation of these criteria is expressed in harassment posts. In particular, we propose the following research question:

$RQ_{\text{consent-violation}}$: How can we leverage natural language to identify consent violations expressed in harassment posts?

Answering $RQ_{\text{consent-violation}}$ will help in understanding an unexplored and misunderstood topic, consent, and its violation through actual stories of online and real-world harassment.

1.2 Contributions and Novelty

We list contributions and novelty for each project below.

1.2.1 iROGUE: Identifying Rogue Behavior from App Reviews

Our work's novelty lies in leveraging app reviews to identify rogue apps and their rogue functionalities. App reviews inspite being public source of information, still remain unexplored in the rogue scenario. We introduce assigning alarmingness scores to reviews and rogue scores to apps, based on the reported rogue behavior. We contribute to mobile app security by providing:

- iROGUE, an app reviews based approach for identifying rogue apps and their functionalities.

  iROGUE consists of three phases. First, iROGUE predicts the alarmingness score of each app review. Second, iROGUE generates a rogue score for each app based on the alarmingness of its reviews. We selected a threshold on rogue score, above which apps are predicted as rogue. Third, iROGUE finds additional rogue apps by examining apps in other datasets of scraped reviews. We envision iROGUE to be incrementally updated by adding alarming reviews, of newly found rogue apps. Moreover, apps that don't have reviews yet, will be identified rogue, when their new reviews arrive.

- A ranked list of rogue apps along with their alarming reviews revealing rogue behavior.

  App platforms such as Apple App Store can prioritize scrutinizing the rogue apps ranked high in the list or ask their developers to work on providing user privacy. Moreover, app users (potential victims) can be warned of the identified rogue apps before installing them on phones.

We discuss about this project in chapter 2.

1.2.2 METHREE: Highlighting Incidents, Effects, and Requested Advice from MeToo Posts

To the best of our knowledge, we are the first ones to study sentence level identification from long MeToo posts. In particular, we make the following contributions.

- To address $RQ_{highlights}$, we curate METHREE, a dataset containing 8,947 sentences, labeled for the three categories. Constructing a sufficiently natural and precise dataset turns out to be nontrivial. We leverage active learning for labeling with tractable manual effort.

- We train a natural language model to identify and inturn highlight these three categories of sentences from long MeToo posts: (i) incident, (ii) effects, and (iii) advice sought. Our approach incorporates modern Natural Language Processing (NLP) techniques to achieve strong results.

We discuss about this project in chapter 3.

1.2.3 Understanding Inciting Speech in Social Media

In the religious setting, inciting speech as a discrete category of antisocial communication has not been computationally identified. Hence, our research is inherently novel. We not only

identify inciting speech but also study its pragmatics through dialogue acts and impoliteness super-strategies.We make three contributions.

- Uksaana, a dataset of 7,000 sentences annotated for three rhetorical strategies: (i) identity, (ii) imputed misdeeds, and (iii) exhortation.
- We develop embedding-based and transformer-based models and choose the best one to identify sentences having rhetorical strategies in a post.
- We identify the characteristics of inciting speech, uncovering dialog acts and impoliteness super-strategies used by the writer of such text.

We discuss this project in chapter 4.

1.2.4 Proposal: Understanding Violation of Consent

Prior studies (Singh 2022; Hohfeld 1923; Koch 2018) focus on formulating the theory of consent, however understanding its violation through actual posts has not garnered much attention. We plan to analyze how some of the nine criteria of consent are violated in such harassment stories. We discuss about this plan in chapter 5.

Table 1.2 shows the timeline for the entire thesis research.

Table 1.2: Proposed plan.

	Task	Status	Estimate Time of Completion
1	iROGUE	Complete	–
2	METHREE	Almost Complete	June 2023
3	UKSAANA	Complete	–
4	Understanding consent	Ideation	Dec 2023

CHAPTER

2

IROGUE: IDENTIFYING ROGUE BEHAVIOR FROM APP REVIEWS

An app user can access information of other users or third parties. We define rogue mobile apps as those that enable a user (abuser) to access information of another user or third party (victim), in a way that violates the victim's privacy expectations. Such apps are dual-use and their identification is nontrivial. We propose iROGUE, an approach for identifying rogue apps based on their reviews, posted by victims, abusers, and others. iROGUE involves training on deep learning features extracted from their 1,884 manually labeled reviews. iROGUE first identifies how alarming a review is with respect to rogue behavior and, second, generates a rogue score for an app. iROGUE predicts 100 rogue apps from a seed dataset curated following a previous study. Also, iROGUE examines apps in other datasets of scraped reviews, and predicts an additional 139 rogue apps. On labeled ground truth, iROGUE achieves the highest recall, and outperforms baseline approaches that leverage app descriptions and reviews. A qualitative analysis of alarming reviews reveals rogue functionalities. App users, platforms, and developers should be aware of such apps and their functionalities and take measures to curb privacy risk.

2.1 Introduction

With the expansion of mobile technologies, privacy threats arise not only from malicious or careless app developers, but also from app users. The privacy expectations of an app user or a third party (i.e., a *victim*) are violated when the victim (1) doesn't know about another user (i.e., an *abuser*) accessing the victim's information (spying) or (2) may know about the access, but is uncomfortable with it. The latter case includes incidents of forced consent or when public information on apps (such as a dating platforms) is accessed beyond the victim's level of comfort, such as profile stalking. We use term *rogue behavior* to mean these two types of information access, and use *rogue apps* to mean the apps that enable rogue behavior.

Research (Chatterjee et al. 2018; Freed et al. 2019; Havron et al. 2019) shows that rogue apps may cause discomfort, fear, and potential harm to the victim. Possible ways to prevent this risk include highlighting these apps and their rogue functionalities to users, warning app distribution platforms, and informing app developers. All these actions rely upon identifying rogue apps and their functionalities. Our proposed approach, iROGUE, shows how to do so.

Previous studies (Chatterjee et al. 2018; Roundy et al. 2020) focus only on the access performed without the victim's knowledge, but do not consider cases when the victim is uncomfortable of the access (of public information) even if aware of it. Chatterjee et al. (2018) use app descriptions to identify intimate partner surveillance (IPS) apps (subset of rogue apps), the apps that someone can use to spy on his or her intimate partner (spouse, boyfriend, or girlfriend). Such IPS apps are dual-use apps, that have a legitimate purpose but are misused for spying. This concept of dual-use apps also applies to the general setting of rogue apps. Since app descriptions of rogue apps indicate only intended legitimate behavior, app descriptions may not be suitable for identify all misuses. We leverage app reviews to identify all misuses of such apps. We observe that the reviews of an app describe rogue functionalities, its misuse (potential and actual), and the privacy expectations of users. Such reviews are evidence of rogue behavior and should be brought to the attention of users, developers, and app platforms.

Example 1 shows three reviews (edited for grammar), taken from the Apple's App Store[1], and are relevant to the rogue behavior. Although our study is based on Apple App Store's reviews, iROGUE can be applied on reviews from other sources, including Google's Play Store[2].

In Example 1, the first review for AirBeam Video[3], addresses the scenario where the app assists a user to access a victim's information without the victim's knowledge. AirBeam Video is a surveillance app to be installed on the abuser's device. Hence, the victim may not be an

[1] `https://www.apple.com/app-store/`
[2] `https://play.google.com/store/apps`
[3] `https://apps.apple.com/us/app/airbeam-video-surveillance/id428767956`

Example 1: Cases Relevant to Rogue Behavior

Fly on the wall!

(for the AirBeam Video Surveillance app[a])

"with this app, i can spy on my family without them knowing it! it's such an awesome app!"

This app basically ruined my family to an extent

(for the Life360 app[b])

"My mother made everyone in the family get this app. She freaks out when the app doesn't do its job because of random obstacles that mess with the location accuracy. Drains the battery and makes my parents paranoid to know where I am at all times. I don't even do any bad stuff, yet years of trust building are being swept away by the ability to spy on the children of a household. If you're a parent I highly recommend you don't get this app because it is extremely uncomfortable to have and it makes parents trust their children less."

Honest

(for the 3Fun: Threesome & Swingers app[c])

"…A lot of the local people I've talked to (Male half of a couple) have been guys who are saying they're part of a couple, and in all reality are single guys just looking to collect pictures. There is no way to report that that is why you are reporting them. It's just a boilerplate report feature. I feel there should be a way for the 3Fun community to point out people for bad behavior like this."

[a] `https://apps.apple.com/us/app/airbeam-video-surveillance/id428767956`
[b] `https://apps.apple.com/us/app/life360-find-family-friends/id384830320`
[c] `https://apps.apple.com/app/id1164067996`

app user but a third party. The second review, for Life360[4], complains about the problem of inappropriate access of user's location by the user's mother. Due to the unequal power dynamics between the victim (reviewer in this case) and the abuser (mother in this case), the victim is forced to install apps that violate privacy. The third review, from 3Fun[5], describes the story of improper access of profile pictures. Even though the profile pictures are public, the victim is uncomfortable with the access. It is common for users to upload such information (pictures in this case) on an app. When doing so, they hold expectations of how other users would access it. Information access, as shown in these three, cases may lead to discomfort, fear, or potential harm (Freed et al. 2019; Havron et al. 2019). Thus, despite such cases of information access

[4] `https://apps.apple.com/us/app/life360-find-family-friends/id384830320`
[5] `https://apps.apple.com/app/id1164067996`

being common, they should be brought in front of app developers and platforms. However, app descriptions don't reveal possibility of a user (victim) to be uncomfortable of such access.

To address victims' privacy expectations, we propose the following research questions:

$RQ_{\text{rogue-apps}}$. How can we identify rogue apps from reviews?

$RQ_{\text{rogue-functionality}}$. How can we uncover rogue functionalities?

Section 2.3 To address **$RQ_{\text{rogue-apps}}$**, we propose iROGUE, an approach that is trained on the deep learning features extracted from 1,884 app reviews. iROGUE includes three phases (described in Section 2.3). First, it assigns an *alarmingness* score to each review. The alarmingness score is used to rate and rank each review according to the claims and severity of rogue behavior. Second, iROGUE identifies rogue apps, based on a *rogue score*, computed by aggregating the alarmingness scores of an app's reviews. The rogue score ranks each identified app, according to the rogue behavior reported in app reviews. Such ranking can be useful for app distribution platforms, such as Apple App Store[6] and Google Play Store[7], to prioritize the scrutiny of identified apps. Third, iROGUE identifies additional rogue apps, by examining apps in the other datasets. To evaluate the performance of iROGUE, we report its precision, recall, and F1 score in identifying rogue apps.

To address **$RQ_{\text{rogue-functionality}}$**, we leverage reviews with the top 10 alarmingness scores and manually analyze them to find their rogue functionalities. We further installed a few rogue apps on an iOS device to verify reported rogue functionalities and include our findings in Section 2.4. We contacted Apple App Store[8]. We shared with them the list of identified rogue apps, along with reviews containing evidence against each app. They told us they will investigate the rogue apps and reach out to developers to rectify apps.

Contributions. Our work's novelty lies in leveraging app reviews to identify rogue apps and their rogue functionalities. We introduce assigning alarmingness scores to reviews and rogue scores to apps, based on the reported rogue behavior. We contribute to mobile app security by providing:

- iROGUE, an app reviews based approach for identifying rogue apps and their functionalities.
- A ranked list of rogue apps along with their alarming reviews revealing rogue behavior.

Organization. The rest of this chapter is organized as follows. Section 2.2 describes our preliminary investigation that shows that app reviews contain evidence of rogue behavior.

[6] `https://www.apple.com/app-store/`
[7] `https://play.google.com/store/apps`
[8] `https://www.apple.com/app-store/`

Section 2.3 describes our proposed iROGUE approach to identify rogue apps, along with its evaluation. Section 2.4 shows the procedure to uncover rogue functionalities of rogue apps. Section 4.7 lists related work on information access in mobile apps. Section **??** concludes this chapter.

2.2 App Reviews Reveal Rogue Behavior

We now describe rogue behavior reported in app reviews.

2.2.1 Seed Dataset

Chatterjee et al. (2018) identify 2,707 iOS apps as potentially IPS. Out of these apps, they confirm 414 apps to be IPS, using semi-supervised pruning.

When we collected our data, 724 of Chatterjee et al.'s 2,707 apps (including 125 IPS) were already removed from the Apple App Store[9], meaning only 1,983 were available. Of these 1,983 apps, 1,687 received at least one review from 2008-07-10 to 2020-01-30, yielding 11.57 million reviews in all. Only 210 of these 1,687 apps were on Chatterjee et al.'s IPS list. Table 2.1 describes our *seed dataset*, which comprises these 1,687 apps and their 11.57 million reviews.

Table 2.1: Details of our seed dataset.

App Type	Apps	Apps w/ Reviews	Reviews
Removed	724	–	–
IPS apps	289	210	190,584
Other apps	1,694	1,477	11,381,377
Total	2,707	1,687	11,571,961

2.2.2 Investigating Reviews

Since the seed dataset contains 11.57 million reviews, it is impractical to manually check each review for rogue behavior. Hence, we sampled app reviews containing at least one keyword related to rogue behavior. To form a set of such keywords, we initialized a set with the words: *spy*, *stalk*, and *stealth*. We queried WordNet (Miller 1995) for synonyms of these words. We

[9] `https://www.apple.com/app-store/`

performed the query operation until we didn't find any new word in the set. The resulting set contained keywords: *spy*, *stalk*, *stealth*, *descry*, *chaff*, and *haunt*. However, we didn't consider *chaff* and *haunt* to be relevant for describing rogue behavior in reviews. Also, *descry* is present in only two reviews, both of which are irrelevant for rogue behavior. To expand the set of keywords, we explored other corpora such as PyDictionary[10] and Thesaurus[11] but did not find synonyms that are widely used in app reviews. Moreover, keywords used in the previous study (Chatterjee et al. 2018), such as *track* and *control* bring many false positive reviews. For example, "I like tracking my distance when I walk with my dog." and "…you can also control the audio of your mac through the app …I can control music tracks without having to touch the computer." are not relevant. Thus, our relevant set of keywords reverts to *spy*, *stalk*, and *stealth*. Table 2.2 shows the occurrence of each keyword, in reviews of the seed dataset. We refer to this set as *our keywords*.

There are 5,287 reviews containing at least one of our keywords. From these 5,287 reviews, we randomly sampled 995 reviews for manual scrutiny. This sample involves 179 apps with between 1 and 237 reviews each.

Table 2.2: Occurrence of each keyword.

Keyword	Review Count
Spy	2,479
Stalk	2,605
Stealth	218
Total Unique	5,287

The first author manually checked 995 reviews for rogue behavior. Out of 995 reviews, we found 402 reviews (of 83 apps in this sample) reporting rogue behavior. Our manual analysis categorize these 402 reviews along the dimensions of *story* and *reviewer*. Based on rogue story, we observe reviews of following two types:

***Rogue Act*:** Reviews describing someone performing a rogue behavior. In such reviews, the reviewer is sure about the app's rogue functionality.

***Rogue Potential*:** Reviews express the possibility of rogue behavior. The reviewer may not be sure of rogue functionality, but identifies risks with the app.

[10] `https://pypi.org/project/PyDictionary/`
[11] `https://pypi.org/project/py-thesaurus/`

Example 2 shows a review for each type of rogue story.

Example 2: Types of Rogue Stories

Rogue Act

"This is a really good app if u want to spy on your spouse I found out my boyfriend was cheating on me great app I recommend this app"

Rogue Potential

"…May work well to spy on the kids by 'accidentally' leaving iPhone in secret place."

We also found three types of reviewers writing rogue stories. First, reviewers who are *victims*: they state their concerns and grievances, including frustration at the loss of privacy. Second, reviewers who are *abusers*: they admit to the rogue behavior and sometimes express their delight in it. Third, reviewers are *third persons*: they report on others misusing the app or the potential to misuse. Example 3 shows a review for each type of reviewer.

Example 3: Categories Based on Reviewer

Victim

"I hate this app so much! My mother is always questioning me and if I delete it she will ground me …No one want their parents to stalk them!!"

Abuser

"I can spy on my child whenever i want its amazing he cant go anywhere without me knowing look."

Third Person

"…I don't feel like parents should track their kids AT ALL. everyone needs a little something called trust and if you don't have it then your kids will act out and have to become sneaky. This app is designed to track families and see everything just like the parent is with you at all times. I do have this app but only with my fiends and we don't stalk each-other we just use it to see where everyone's at. And Bc we are so close and we all wanted it we all got it."

Table 2.3 shows the count of stories for each type of reviewer. The third person writes most

of the potential reviews (44 out of 47) because such cases are only possibilities and not acts, meaning the reviewer is neither a victim nor an abuser. Whereas, abusers write other three potential cases. Such reviews (by abusers) indicate possible threats with the reviewed app, but suggest other apps for better rogue functionalities. For rogue act reviews, we found abusers (219) and victims (120) writing a majority of stories, followed by third person (16).

Table 2.3: Count of stories for each reviewer type.

Reviewer	Rogue Act	Rogue Potential
Victim	120	0
Abuser	219	3
Third Person	16	44

To sum up, reviews describe apps' rogue behavior and show how victims such as children, parents, and friends are abused.

2.3 The iROGUE Approach

iROGUE consists of three phases. First, iROGUE predicts the alarmingness score of each app review (Section 2.3.1). Second, iROGUE generates a rogue score for each app based on the alarmingness of its reviews. We selected a threshold on rogue score, above which apps are predicted as rogue (Section 2.3.2). Third, iROGUE finds additional rogue apps by examining apps in other datasets of scraped reviews (Section 2.3.3). Figure 2.1 shows an overview of the iROGUE approach. We envision iROGUE to be incrementally updated by adding alarming reviews, of newly found rogue apps. Moreover, apps that don't have reviews yet, will be identified rogue, when their new reviews arrive.

2.3.1 Computing Alarmingness of Reviews

Section 2.2 shows that app reviews reveal evidence of rogue behavior. However, identifying evidence in reviews is nontrivial, especially when an app receives a large number of reviews. Instead of binary classification of reviews, we introduce the alarmingness score that not only identifies relevant reviews but also ranks them based on the rogue behavior. To assess alarmingness of a review, we consider two factors: (i) the review's *convincingness* about rogue behavior and (ii) the *severity* of the reported rogue behavior. The alarmingness score of a review is the

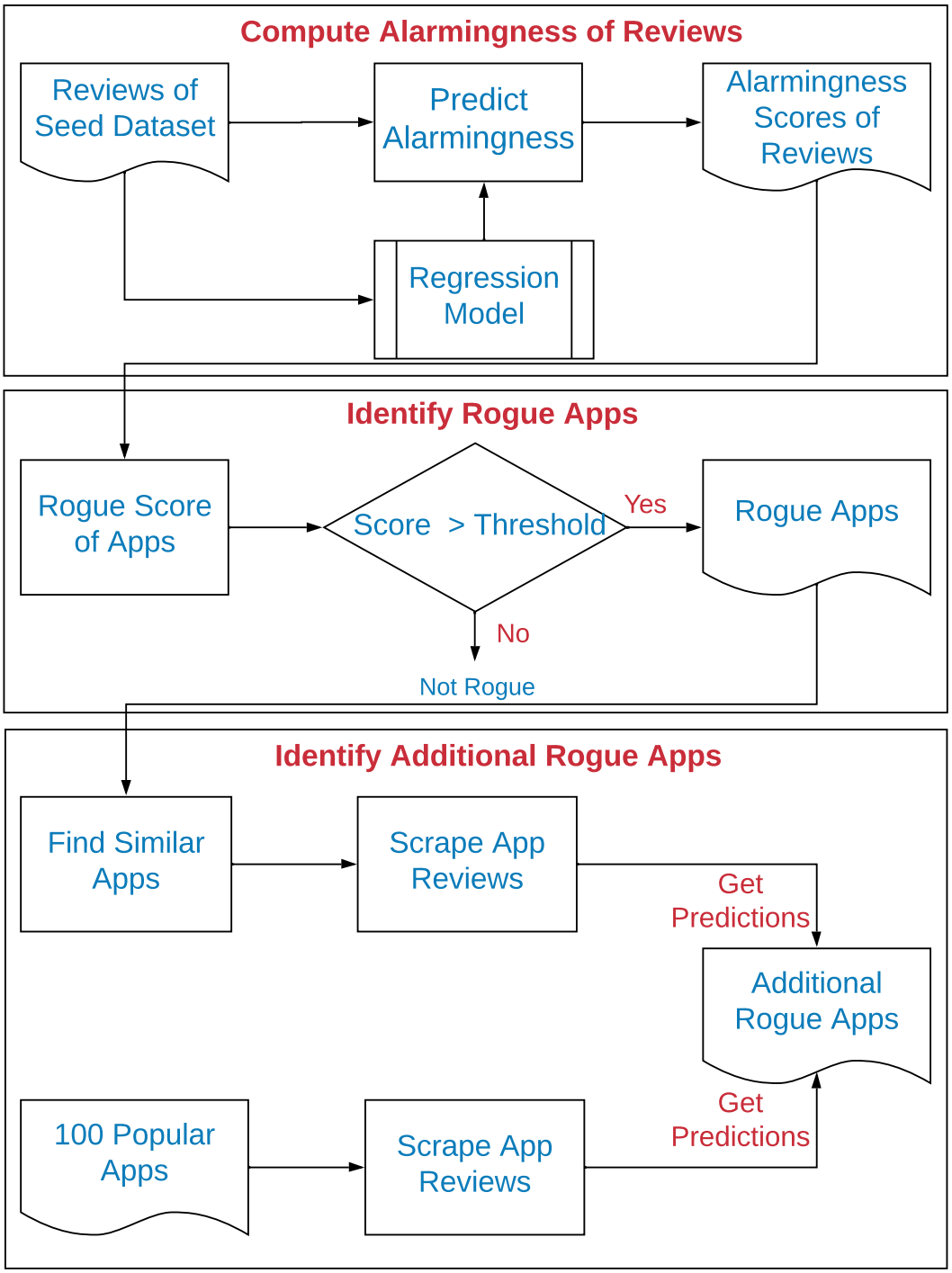


Figure 2.1: Overview of iROGUE approach.

geometric mean of its convincingness and severity scores.

Reviews can vary in their claims about the rogue behavior. Some reviews report detailed rogue behavior, whereas some others are merely suspicion. The convincingness score measures how convincing the app review is in describing the rogue behavior. In Example 4, the first review is unrelated to rogue behavior and hence is not convincing. The second review describes the reviewer's suspicion on the app, which may or may not be true (slightly convincing). The third review (by an abuser) confirms the rogue behavior but lacks details of the rogue functionalities or victims. On the contrary, other reviews (in Example 4) are extremely convincing because they confirm rogue behavior along with mentioning the location feature, or how to set up devices, or the victims being stalked. Extremely convincing reviews include cases when the app is used for positive purposes (tracking family members or pets for safety) but has the potential to be misused in future. The reviews that are slightly, moderately, or extremely convincing are relevant to identifying rogue behavior. Assigning a convincingness score helps in ranking these reviews according to the strength of their claims.

The severity score measures the effect of rogue behavior on the victim. Example 5 shows range of reviews varying in severity. The first review is unrelated to rogue behavior. Thus, it is not severe. The second review shows that the rogue act is performed with consent, making this review a slightly severe case. The third review is written by the abuser and lacks the victim's perspective to analyze rogue effect. We assume such acts are performed without consent and consider them moderately severe. The fourth review describes the victim's misery. The victim even says "This app has truthfully ruined my teenage years" in the review, which gives solid evidence to be an extremely severe case. Moreover, in the fifth review, the victim complains that others can see when he was last active (also known as last seen information). This is the public information on each profile, but still the victim is uncomfortable with the access. App developers should be aware of such users' privacy expectations. However, such cases are still missed by the existing studies (Chatterjee et al. 2018; Roundy et al. 2020). Since app reviews discuss privacy expectations, we are able to identify such cases and rate them extremely severe.

We first rate the convincingness and severity of rogue behavior reported in the app reviews (Section 2.3.1). We then extract deep learning features from the reviews (Section 2.3.1). Leveraging the annotated set and extracted features, we evaluate various regression models and choose the best one (Section 2.3.1). Finally, we calculate the alarmingness score of an app review as the geometric mean of its predicted convincingness and severity scores.

Review Annotation

We selected 1,884 reviews from the seed dataset: 952 (set s_1) that contain at least one of our keywords and 932 reviews (set s_2) that do not contain any of those keywords. While preparing

Example 4: Varying Degree of Convincingness

1: Not Convincing

"It is such a great game, love it so much!"

2: Slightly Convincing

"Setup was a breeze. Quicktime 7 pro found it easily. Unfortunately, resolution seems much, much, lower than hoped. Video size can not be adjusted live. Hate to be a hater. May work well to spy on the kids by 'accidentally' leaving iPhone in secret place."

3: Moderately Convincing

"This app is perfect for stalking people…"

4: Extremely Convincing

"This app is awesome for our family to keep track of where everyone is at all times! (You can turn the location off too in case you want to be in stealth mode when buying Christmas presents too.) …Even our dog knows that the alert sound when a family member arrives home means …"

"…I use it to spy on my dogs while I'm at work; so I use it for fun, nothing fancy. My iPad is my camera, and my iPhone is my viewer. …"

"bro this app is high key creepy. when i'm with my dad on his days my mom even mentions how she knew everything i was doing and it even made my dad creeped out. if you need this app then ngl yo wack. i don't want my mom stalking me."

this annotation data, we exclude the reviewers' identifiers such as their usernames. Each selected review is rated for convincingness and severity on a four-point Likert scale (1: not, 2: slightly, 3: moderately, 4: extremely). For quality of annotations, we measure Inter Rater Reliability (IRR) via Intraclass Correlation Coefficient (ICC) (Hallgren 2012). ICC is suitable for Likert scale ratings. Unlike other IRR measures such as Cohen's (Cohen 1988) kappa, which are based on (all or nothing) agreement, ICC takes into account the magnitude of agreement (or disagreement) to compute IRR.

The annotation was conducted in three steps. First, two authors rated 599 of 1,884 selected reviews according to the initial set of annotation instructions. The initial instructions included definitions (of convincingness and severity scores) and examples corresponding to each point on the likert scale. In this step, for each annotator, we calculated the alarmingness scores of

Example 5: Varying Degree of Severity

1: Not Severe
"Love the graphics so far it is a great game"

2: Slightly Severe
"I love this app, just great because you can time your day accordingly, I like my girlfriend knowing where I am and I love stalking her, we have fun with it.. . . "

3: Moderately Severe
"This app is perfect for stalking people. . . "

4: Extremely Severe
"honestly if you want your kid to rebel against you even more, this is the app for you! This app has truthfully ruined my teenage years all because my mother now has a way of tracking me down 24/7. I couldn't do the normal teenage things because I was being stalked all day . . . "

". . . i want to share my last seen just to my family and my girlfriend not others. please add new feature in privacy that i can share my last seen to no body except my family and girl friend thanks soo much !"

reviews using the convincingness and severity ratings. If the alarmingness scores computed for both the annotators were not at least three (median value on Likert scale), or both are not less than three, annotators resolved such cases via discussions. After discussing, the annotators produced the final set of annotation instructions. In the second step, the annotators followed the final instructions and rated 900 reviews. In this step of the annotation process, we achieved ICC of 0.9195 for convincingness and 0.9190 for severity. An ICC score in the range 0.75–1 indicates excellent agreement (Hallgren 2012). In the third step, the remaining reviews were divided among the two annotators so that only one annotator rates each review.

For reviews that were rated by the two annotators, we computed the average convincingness and severity scores. This annotation study possessed minimal risk and was approved by the Institutional Review Board (IRB) of our university.

Extracting Deep Learning Features from App Review

We obtained the feature vector of each app review as follows:

Combine Sentences: Remove periods in each app review and combine all its sentences to form single sentence.

Text Preprocessing: Remove all punctuation marks, stop words (Uysal and Gunal 2014), and our keywords, the latter because those keywords may correlate with reviews with higher scores and could create bias in the model.

Sentence Embedding: We leverage the Universal Sentence Encoder (USE) (Cer et al. 2018) to extract embeddings for each app review. USE uses Deep Averaging Network (DAN) to provide a 512-dimension embedding for a long text. USE is trained on a large variety of natural language tasks with the aim of capturing the context. In our case, USE directly provides sentence level embeddings of an app review, by keeping the context intact. However, alternatives such as GLoVe (Pennington et al. 2014) and Word2Vec (Mikolov et al. 2013) lose such context. We leverage the pretrained USE network by using Tensorflow Hub[12].

Training Regression Model

We treat score prediction as a multi-target regression problem (Borchani et al. 2015). Here, the 1,884 annotated reviews form the training set, and convincingness and severity are target variables, predicted using the extracted deep learning features.

We evaluated the performance of three regression models: support vector regression (Basak et al. 2007), random forest (Smith et al. 2013), and decision tree (Xu et al. 2005), by ten-fold cross validation on our dataset. To mitigate bias of our keywords, we remove such keywords in the preprocessing step, so that the regression model learns from the the context of the review and not from specific keywords. Table 2.4 shows average and standard deviation of mean squared error (MSE) (Sammut and Webb 2010) in ten folds. The reported MSE is the combined MSE for two targets. The Support Vector Regressor (SVR) yields the smallest MSE, so we choose it for the subsequent phases of our approach.

Table 2.4: Performance of three regression models on ten-fold cross validation.

Regression Model	Average MSE	Standard Deviation
Decision Tree	1.344	0.402
Random Forest	0.712	0.417
Support Vector	**0.625**	0.458

We use this trained model to predict convincingness and severity scores of all 11.57 million reviews in the seed dataset. The alarmingness of each review is calculated by taking the geo-

[12] `https://www.tensorflow.org/hub`

metric mean of its predicted convincingness and severity. We use geometric mean because it ensures high alarmingness value only if both the convincingness and severity scores are high.

2.3.2 Identifying Rogue Apps

We produce an app's rogue score by aggregating the alarmingness scores of its reviews as follows.

Weighted Mean of Alarmingness: In general, for a rogue app, a small proportion of reviews report rogue behavior. Thus, we need to catch rogue apps using their few reviews that have high values on the alarmingness scale. Thus, we assign weights to reviews based on their alarmigness, as follows:

Defining score buckets: While annotating reviews, we defined levels of convincing reviews (not convincing to extremely convincing) and severe reviews (not severe to extremely severe) on a Likert scale. We also follow same levels on the alarmingness scale (1: not alarming to 4: extremely alarming). We define a score bucket between every consecutive level of alarmingness (not alarming to slightly, slightly alarming to moderately alarming, moderately alarming to extremely alarming). Table 2.5 shows how score buckets are formed using levels of alarmingness.

Assigning weights to score buckets: We have 11.57 million reviews in the seed dataset. Based on the alarmingness computation (Section 2.3.1), we calculated the probability of a review falling in a score bucket. Since, the reviews reporting rogue behavior are less, probabilities in buckets 2 and 3 are less than that in bucket 1. We take inverse of these probabilities to get the weights for each score bucket. As a result, we assign higher weights to buckets 2 and 3 than to the bucket 1. Table 2.5 also shows the weight assigned to each score bucket.

Table 2.5: Score buckets for alarmingness.

Alarmingness Score Range	Alarmingness Level Range	Bucket	Bucket Weight
$[1,2)$	Not alarming to Slightly	1	2.29×10^{-3}
$[2,3)$	Slightly to Moderately	2	6.08×10^{-2}
$[3,4]$	Moderately to Extremely	3	9.36×10^{-1}

If a_1, a_2, ..., a_n are the alarmingness scores of an app's reviews, and w_1, w_2, ..., w_n are their respective weights (according to Table 2.5), then, $W_{\text{alarmingness}}$, the weighted mean of alarmingness is given by:

$$W_{\text{alarmingness}} = \frac{a_1 * w_1 + a_2 * w_2 + \ldots a_n * w_n}{w_1 + w_2 + \ldots w_n}$$

The weighted mean of alarmingess ranges from 1 to 4.

Normalized Count: The weighted mean of alarmingness does not account for the count of reviews that report rogue behavior against an app. Suppose, *app A* has 15 reviews reporting rogue behavior and *app B* has 25 reviews reporting rogue behavior. If all reviews reporting rogue behavior have the same alarmingness score, the weighted means of the two apps would be the same. But, *app B* shows more evidence of rogue behavior and should have a higher rogue score than *app A*. Thus, we also consider the count of reviews. For each app, we calculate the number of reviews in bucket 3. We tried incorporating counts of other buckets, but it led to worse performance of the approach.

The minimum possible value of the count is zero. However, in some cases, counts can be high, leading to no definite upper limit. Thus, we normalize the counts of all the apps between one and four.

We want to assign high rogue score to apps that have high scores in both (i) weighted mean of alarmingness and (ii) normalized count. Thus, rogue score is computed as the geometric mean of these two values.

Selecting Threshold for Prediction

For each app in the seed dataset, iROGUE computes the rogue score. All apps are ranked in decreasing order of rogue scores. The apps with a score greater than a threshold are predicted rogue. To decide the correct threshold, we follow two steps: (i) label the ground truth of rogue apps and (ii) vary a threshold between certain values and choose the threshold which gives the best performance of iROGUE.

We create our ground truth by manually scrutinizing reviews. However, scrutinizing reviews of all 1,687 apps (seed dataset) is not feasible. Thus, first, we scrutinize the 50 most alarming reviews (with minimum score of two—at least slightly alarming) for apps with the highest 100 rogue scores. Second, we scrutinize reviews containing our keywords for these 100 apps. The first step is aligned with our approach since it checks top alarming reviews. However, the second step is neutral because it searches for evidence for the apps that iROGUE failed to

identify through top alarming reviews. This way we mitigate the threat of bias, while curating ground truth for apps with the highest 100 rogue scores.

We label an app as rogue provided any of the scrutinized reviews report a rogue behavior. Table 2.6 shows the types of reviews we consider indicative or otherwise of a rogue evidence. If the reviews of an app describe information access performed without the victim's knowledge or when the victim shows discomfort (first three reviews in Table 2.6), we consider the app as rogue. Also, some rogue apps are used for positive purposes such tracking family members for safety (fourth review in Table 2.6) but still possess a potential for future misuse. Similarly, apps used for tracking pets or other objects are considered rogue (fifth review). Reviews of some apps don't possess any misuse in present or in future, leading to their final label of not rogue (sixth review). Through manual inspection, we determine that of the 100 apps, 73 are rogue and 27 are not.

For the 100 apps, the rogue score varies between 1.74 and 3.60. We vary the threshold from 1.73 to 3.59 in steps of 0.01. Apps with a rogue score above the threshold are predicted rogue but not rogue otherwise. At each value of threshold, we report the recall, precision, and F1 score.

Table 2.7 shows the performance achieved at specific thresholds. As we increase the threshold, the precision increases at the cost of recall. For rogue apps, a false negative costs more than a false positive because a false negative leaves a rogue app undetected, which can harm many victims, whereas a false positive causes only wasted effort in manual scrutiny. Hence, achieving high recall is more important than achieving high precision. Thus, from Table 2.7, we choose 1.73 threshold that gives the best recall of 100% at 73% precision. Since we fine-tune the threshold on the same seed dataset, we also check iROGUE's performance (using the chosen threshold) on the other dataset (Section 2.3.3).

Performance of Baseline Methods

On the seed dataset, we also check the performance of baseline methods described below.

Our Keywords on App Description. We search for the presence of one of our keywords (*spy*, *stalk*, and *stealth*) in app descriptions. Apps whose descriptions contain any of these keywords are predicted rogue, whereas other apps are predicted as not rogue.

Extended Keywords on App Description. We identify additional relevant keywords by extracting verbs through Part-Of-Speech (POS) tagging (Manning 2011). POS tagging marks every word in a sentence to an appropriate part of speech (verb, noun, adjective, and so on). Applying this process on descriptions of 73 rogue apps (from the ground truth) produced 145 verbs, out of which six (*track*, *monitor*, *locate*, *control*, *stolen*, *lost*) we selected as relevant

Table 2.6: Instructions followed for labeling rogue apps.

Type of case	Subtype of case	Example	Evidence of rogue behavior
Tracking people's information	Without the victim's knowledge	*"Now that I can spy on my wife I will always know when she is cheating"*	Yes
Tracking people's information	With the victim's knowledge but with discomfort	*"Ok my mom got this for me and . . . it's kinda creepy that this app was made so parent could basically stalk their kids."*	Yes
Tracking people's information	Public information but the victim is uncomfortable	*"I had someone cyberstalking and harassing me. Multiple attempts in every way shape and form were made to contact app-name to block and ban the stalker's account due to a concern for my well- being."*	Yes
Tracking people's information	Positive purpose	*"I love finding my family members. Wife was in bad car wreak and I was able to find her location using this app. Thank you!"*	Yes
Tracking pets or other objects		*"Wow! Day one and I'm stalking my puppet like a soccer mom that ran out of adderall! I'm very excited to use this to interact with my puppet while I'm at work and to check in on the dog walker!"*	Yes
Not related to information accessing		*"an absolutely amazing and very helpful app. i don't know how i would keep track of prayer times without it. love the app. thank u!!!"*	No

to rogue behavior. The verbs: *stolen* and *lost* are relevant because they describe the apps that are used to find a misplaced phone, which indicates an ability to track another device. We extend our keywords by adding these six verbs. Apps whose description contain these

Table 2.7: Choosing an appropriate threshold according to the recall scores.

Threshold	Precision (%)	Recall (%)	F1 Score (%)
1.73	**73.00**	**100.00**	**84.39**
1.74	73.40	94.52	82.63
1.75	76.13	91.78	83.22
1.76	76.82	86.30	81.29
1.77	76.92	82.19	79.47
1.78	78.37	79.45	78.91
1.79	79.71	75.34	77.46
1.80	80.95	69.86	75.00
1.81	80.95	69.86	75.00
1.82	81.96	68.49	74.62
1.83	81.03	64.38	71.75
1.84	80.35	61.64	69.76
1.85	82.69	58.90	68.80
1.86	83.33	54.79	66.11
1.87	82.97	53.42	65.00

keywords are predicted rogue.

T% Keyword Reviews. For each app, we compute the percentage of reviews containing our keywords. We set a threshold, T, on this percentage, above which apps are predicted rogue. In our evaluation, T takes the values of 0.3, 0.2, and 0.1, respectively.

Table 2.8 summarizes the precision, recall, and F1 scores of all baselines and our approach. Our keywords on the description predict only one rogue app, leading to 100% precision (highest among all). However, our keywords miss 72 rogue apps, which leads to the worst recall of 1.36%. Among all the baselines, keyword search on reviews with 0.1% threshold achieves the highest recall of 65.07%, which is much lower than iROGUE's recall value. iROGUE's better performance may be due to fine tuning iROGUE's threshold on the same seed dataset. Thus, we also compare iROGUE's performance with these baselines on the other dataset (Section 2.3.3).

Examples 6 and 7 show alarming reviews of Find My Family & Friends App[13] and OurPact Jr. Child App[14], which iROGUE correctly identifies as rogue. Both of them are dual-use apps. Find My Family & Friends is a safety app, but alarming reviews report parents misusing the tracking functionality on children, to which children are uncomfortable. Moreover, the alarming reviews of the OurPact Jr. Child App report that parents can monitor children's texts and visited websites

[13] `https://apps.apple.com/us/app/life360-find-family-friends/id384830320`
[14] `https://apps.apple.com/us/app/id1127917970`

Example 6: Rogue App from Seed Dataset

App: Find My Family & Friends[a]
Rogue Score: 3.60/4.00

Alarming Review 1
Alarmingness: 4.00/4.00
Date of Review: 2019-11-28
"...Such a terrible thing for unaware parents to use. Most parents think teens don't need privacy and they constantly need to know where they are and what they're doing and who they're with at all times. This may make the parent feel at peace but what about the child? It's selfish of parents to not take into consideration of how the teen may feel about always having this app and the parent giving them a very stalkish feeling, it's very uncomfortable."

[a] `https://apps.apple.com/us/app/life360-find-family-friends/id384830320`

Table 2.8: Performance (in %) of baseline methods and iROGUE on the seed dataset. Bold value for a metric indicates the highest score among all approaches.

Method	Recall	Precision	F1
Our keywords on app descriptions	01.36	**100.00**	02.68
Extended keywords on app descriptions	61.64	80.35	69.76
0.3% keyword reviews	46.03	96.66	62.36
0.2% keyword reviews	50.79	96.96	66.66
0.1% keyword reviews	65.07	95.34	77.35
iROGUE	**100.00**	73.00	**84.39**

by installing the app on the child's device. In Section 2.4, we discuss these rogue functionalities in detail.

2.3.3 Identifying Additional Rogue Apps

The scoring part of our approach is not dependent on the choice of candidate apps and could be applied on any dataset of apps. To identify additional rogue apps, we applied iROGUE's first two phases on two datasets: (i) dataset of similar apps and (ii) dataset of 100 popular apps in the utilities category.

Similar Apps

We retrieved 975 apps (similar to 100 predicted apps from the seed dataset), using the Apple App Store's recommendations ("You May Also Like" section). Our motivation in using Apple's recommendations is that these apps should offer functionalities similar to those in 100 predicted apps. Out of the 975 apps, reviews of 896 apps were present on the Apple App Store, over the period 2008-08-13 to 2022-08-24. We obtained 2,652,678 reviews. These 896 apps along with their reviews form our *snowball dataset*, as shown in Table 2.9.

Table 2.9: Apps and reviews in the snowball dataset.

Similar Apps	Reviews
896	2,652,678

We apply iROGUE's first two phases (described in Sections 2.3.1 and 2.3.2) on the snowball dataset. iROGUE predicts 138 rogue apps. Examples 8 and 9 show alarming reviews of two such apps from the snowball dataset, Smart Family Companion App[15] and Bark - Parental Controls app[16].

In the snowball dataset, to curate the ground truth, we follow the same labeling process as described in Section 2.3.2, for the apps with the highest 200 rogue scores. That's how we label 132 apps as rogue.

Table 2.10 shows the performance of all baseline methods and iROGUE on the snowball dataset. Our keywords when used on descriptions predicts only one app as rogue, leading to the lowest recall. This is because, on Apple App Store[17], dual-use apps are not advertised using keywords: *spy*, *stalk*, and *stealth*. The same approach achieves 100% precision but high recall is desirable in the context of rogue apps.

On app descriptions, extended keywords perform better (68.18% recall at 85.71% precision) than our keywords due to commonly used words (such as *track*, *locate*) in app descriptions. Our keywords are applied on reviews (rows 3–5 in Table 2.10), and discover evidence of rogue behavior. However, among all approaches, iROGUE yields the best recall of 77.27%. As we discussed, high recall is desirable than high precision, we conclude that iROGUE outperforms all other methods.

[15] `https://apps.apple.com/us/app/smart-family-companion/id1352914754`
[16] `https://apps.apple.com/us/app/id1477619146`
[17] `https://www.apple.com/app-store/`

Table 2.10: Performance (in %) of baseline methods and iROGUE on the snowball dataset. Bold value for a metric indicates the highest score among all approaches.

Method	Recall	Precision	F1
Our keywords on app descriptions	00.75	**100.00**	01.48
Extended keywords on app descriptions	68.18	85.71	**76.26**
0.3% keyword reviews	41.66	91.66	57.29
0.2% keyword reviews	44.69	92.18	60.20
0.1% keyword reviews	51.51	88.31	65.07
iROGUE	**77.27**	73.91	75.55

100 Popular Utility Apps

Surveillance apps that can be misused for spying fall under the "Utilities" category, making utilities an important category to scrutinize. We consider 100 popular utility apps that are mentioned on Apple App Store page[18]. Out of 100 apps, nine are already scrutinized either in the seed or snowball dataset. For the rest 91 apps, we retrieved 392,928 reviews, over the duration of 2008-10-18 to 2022-08-04.

iROGUE predicts only one app as rogue, which after reviews' scrutiny by us, comes out to be non-rogue. We also scrutinize 10 apps with the highest rogue scores, by reading their top 50 alarming reviews and reviews containing our keywords. But, none of them are actually rogue. Since the Apple App Store[19] contains a wide variety of utility apps, the selected 91 apps contain subcategories such as payment, calculator, and television remote. As a result, no video surveillance or parental control app, which have high potential to be misused, are part of 91 apps. Thus, popular utility apps don't form a good candidate set for rogue apps. This problem can arise for any generalized set of apps under any category. Thus, an iterative process of checking similar apps (through iROGUE), to the already identified rogue apps, can accelerate identifying more and more rogue apps.

2.4 Uncovering Rogue Functionalities

We now uncover rogue functionalities that are found via app reviews. Identifying such functionalities can help both users and developers. App users can understand the risk associated

[18] https://apps.apple.com/us/genre/ios-utilities/id6002
[19] https://www.apple.com/app-store/

with the app and developers can rectify apps to reduce such risks.

For this study, we considered apps in the seed dataset with the 40 highest rogue scores. For each app, we manually analyzed its description and its 10 most alarming reviews discovered by iROGUE. An app's description provides basic knowledge about the app's functionalities and alarming reviews report misuse of such functionalities. Through this exercise, we discovered the following types of rogue functionalities:

Monitoring phone activities. Some apps monitor a victim's phone activities, such as browsing history and text messages. Such apps are installed on the victim's device and activities can be monitored on another synced device.

Audio or video surveillance. Some apps enable audio or video surveillance without the victim's knowledge. These apps listen, view, or record a victim's voice or actions and some of them need not be installed on the victim's phone.

Tracking location. Some Global Positioning System (GPS) apps enable tracking a victim's phone, with (forced consent) or without their knowledge.

Profile stalking. Some apps are misused for stalking of user profile or user content (such as images), making the victim uncomfortable of the information access.

Table 2.11 shows these four types of rogue functionalities, and alarming reviews reporting them. Some reviews in Table 2.11 are old (2014 or 2012), but we confirmed that similar concerns are being raised in the recent reviews of the same apps. For example, the Find My iPhone[20] app still lets its users see the location of the connected devices. Due to unequal power dynamics, the victim can be forced to connect to such apps and allow their device to be located.

We also verified the rogue behavior of the SaferKid Text Monitoring App by installing it on two devices: a parent's device (iOS version 14.4.1) and a child's device (Android version 11.0). Activities on the child's device can be monitored on the synced parent's device. Moreover, on the SaferKid app, rogue functionalities such as monitoring text messages, web history, and call history. We verified each of these functionalities. Apps such as SaferKid are advertised as safety apps for children but can be secretly or forcefully installed on another device to monitor the user's activity. Not only parents, but any individual can misuse such apps by installing them on the victim's phone.

[20] `https://apps.apple.com/us/app/find-my-iphone/id376101648`

Table 2.11: Types of rogue functionalities.

Rogue Functionality	App Example	Alarming Review
Monitoring phone activities	SaferKid Text Monitoring App[21]	*…Tracking things like social media, texts, and search history is just a complete disregard of privacy. You have to have trust in your kids …Apps like these shouldn't be allowed. IF YOU TRUST YOUR KID, DONT DOWNLOAD.* (Date: 2019-12-07)
Audio or video surveillance	Find My Kids: Parental control[22]	*This app proves to have a invasion of privacy. Due to the fact if your kids was at a friends house and talking to his friends parents, this app records what is going on and is a invasion of privacy. If your child left their phone downstairs or anywhere and they are playing it can record private conversation between adult and is a unsafe …* (Date: 2019-01-16)
Tracking location	Find My iPhone[23]	*…It's supposed to be used to recover a lost phone, not to religiously stalk your children.… The fact that a mom actually installed this app onto her son's phone without his knowledge is flat out wrong. …If you're constantly monitoring your child 24/7, just imagine what your child will do when they go off to college. …* (Date: 2014-02-13)
Profile stalking	WhatsApp Messenger[24]	*…However there is one negative about the App! The stalker look at the time stamp to monitor other people not nice please improve on that we need a sense of privacy from theses stalker"* (Date: 2012-11-21)

2.5 Related Work

We describe previous works focusing on (i) spying through mobile apps, (ii) user privacy on social media, and (iii) NLP techniques to find apps' privacy issues.

2.5.1 Spying through Mobile Apps

Prior studies (Chatterjee et al. 2018; Roundy et al. 2020; Tseng et al. 2020; Freed et al. 2019, 2018; Zou et al. 2021; Tseng et al. 2021) investigate how technology is abused for spying. A major segment of this research deals only with IPS. Chatterjee et al. (Chatterjee et al. 2018) identify IPS apps with carefully designed search queries and manual verification based on app information. They leverage information such as app descriptions and permissions. However, for dual-use

apps, the actual usage deviates from the intended purpose shown in app descriptions. To identify such misuse, we focus on the evidence provided in app reviews. Moreover, the scope of rogue apps is broader than IPS apps.

Roundy et al. (Roundy et al. 2020) focus on identifying apps used for phone number spoofing and message bombing, which lie outside the scope of rogue apps. Conversely, rogue apps include those that enable stalking public information, which are outside their scope. Roundy et al. use metadata such as installation data, to uncover spying apps that are installed on infected devices. However, we focus on evidence of rogue behavior present in app reviews, to uncover rogue apps. Roundy et al. rely upon Norton's security app (NortonApp) to determine which devices are infected. Thus, their approach would miss apps that a general user can leverage to spy.

Some prior studies focus on analyzing spyware apps or victims' experiences. Freed et al. (Freed et al. 2019) present a qualitative analysis of victims' experiences, including their technology-related concerns. They report that security vulnerabilities were present in the phones of 14 out of 31 victims in their sample. Tseng et al. (Tseng et al. 2020) study the IPS problem from the attackers' perspective. They analyze online forums in which attackers participate, propose a taxonomy of IPS tools and attacks. Tools may require physical device access, e.g., to install GPS trackers; or, they may rely on virtual access, e.g., through shared accounts of intimate partners. Attacks may include coercion or subterfuge, or may involve hiring another person to spy on someone. Havron et al. (Havron et al. 2019) propose a consultation method, called clinical security, to help victims by discovering and removing spyware, and advising victims about security vulnerabilities in their phones. Moreover, Tseng et al. (2022) develop sociotechnical systems with feminist notions to help IPV survivors. Freed et al. (Freed et al. 2018) survey spyware apps for intimate partners. They mention covert apps (also known as dual-use apps) that are capable of spying on victims but are not advertised as such. Fassl et al. (2022), through app reviews, study users' expectations from anti-stalkerware apps. They perform thematic analysis on 518 reviews of two apps and find a huge gap between users' perception and the actual abilities of such apps. All these studies along with others (Bellini et al. 2021; Zou et al. 2021; Tseng et al. 2021) are limited to IPS apps and not the broader set of rogue apps. Moreover, they do not consider cases when the victim is uncomfortable of the access (of public information) even if aware of it.

2.5.2 User Privacy

Prior works study risk of losing users' private information on online social media platforms and propose methods to mitigate such risk. Georgiou et al. (Georgiou et al. 2017) protect users'

privacy by giving warnings whenever a user may reveal sensitive attributes such as location or race present in social media posts. Mahmood and Desmedt et al. (Mahmood and Desmedt 2012) claim that the Facebook friends of a user can access the user's private information in a cloaked manner. The study shows that it is possible to stalk and target victims on Facebook. Mahmood and Desmedt et al. provide strategies to avoid such attacks. Reichel et al. (Reichel et al. 2020) study the privacy perspective of users in developing countries. They interview 52 social media users in South Africa to understand their privacy beliefs. Reichel et al. conclude that many participants are concerned about other users being able to see their online posts and messages, instead of the private data collected by the app platform itself. Many participants admitted that unknown people (on WhatsApp and Facebook) stalked or harassed them. To combat these challenges, Reichel et al. provide recommendations to fulfill users' security needs in resource-constrained situations. Some studies contribute to uncovering privacy risks associated with shared images (Henne et al. 2013; Bo et al. 2014; Perez et al. 2017), which can contain bystanders (persons who are not prime subject of image) and are shared widely without bystanders' consent. Hasan et al. (Hasan et al. 2020) leverage visual features to detect bystanders in images present in the Google open image dataset (Kuznetsova et al. 2020).

These studies are applicable to specific social media platforms and not to all rogue apps. Moreover, they do not provide a framework to identify rogue apps and their functionalities.

2.5.3 Using Natural Language Processing

We present prior studies that apply NLP techniques for security and privacy of mobile apps.

Some previous works leverage app reviews and privacy policies to identify user's security and privacy issues. Nguyen et al. (Nguyen et al. 2019) train a classifier to predict if an app review pertains to security and privacy concerns. Using regression analysis, they show that security and privacy related reviews play an important factor in predicting privacy related app updates. Besmer et al. (Besmer et al. 2020) leverage app reviews to understand how users' perception of privacy is reflected in their sentiments about the app. They train a machine learning classifier to determine whether a review is privacy related. Further, they analyze the sentiments of reviews predicted as privacy related. Harkous et al. (Harkous et al. 2018) propose a privacy-centric language model to extract useful information from long privacy policies. The extracted information helps users understand how apps collect and manage users' personal information. To train the language model, they leverage 130,000 privacy policies. The trained model extracts both high-level and fine-grained details from policies. However, these studies focus on how an app which can steal a user's information. In contrast, we focus on the privacy of a victim (user or third party) with respect to another user.

Some prior works distinguish between the actual and the expected behavior (from user's perspective) of an app, by using textual sources such as descriptions and privacy policies. Gorla et al. (Gorla et al. 2014) identify which apps deviate from their descriptions, by extracting topics from app descriptions, using Latent Dirichlet Allocation (LDA), and clustering apps based on those topics. For each cluster, Gorla et al. find outliers with respect to apps' APIs usage. Qu et al. (Qu et al. 2014) and Pandita et al. (Pandita et al. 2013) use NLP techniques on app descriptions and find disparities between app descriptions and functionalities. Zimmeck et al. (Zimmeck et al. 2017) propose an automated system to find Android apps' compliance with their privacy policies. They combine static code analysis and machine learning to uncover inconsistencies between privacy policies and app source code. Out of 9,050 apps, Zimmeck et al. find that 17% of apps collect sensitive information such as location, but do not mention it in their privacy policies. All these works address expectation violation when the app developer has malicious intentions. However, in our work, we address expectation violation when an app user has malicious intentions to spy or stalk. To the best of our knowledge, iROGUE is the first automated system to identify rogue apps.

2.6 Discussion

We proposed iROGUE, an approach to automatically analyze app reviews for detection of rogue apps and rogue functionalities. iROGUE, first, predicts alarmingness of reviews, followed by rogue score for each app. In total, iROGUE predicts 239 rogue apps (100 and 139) from multiple sources, leading to the best recall, as compared to other baseline methods. We have also shared the identified rogue apps along with their reviews, to the Apple App Store. The platform will investigate these apps and will reaching out to the developers for correcting functionalities in their apps.

Below, we describe our data availability, threats to validity, and promising future directions.

2.6.1 Threats to Validity

We now discuss the threats we identify in our work. The identified threats are of two types: (i) the threats that we mitigate; and (ii) the threats that still remain.

Threats Mitigated

We mitigated the following threats to validity. First, reviews in the set s_1 contained our keywords. This may create a bias in the model to predict high scores for only reviews having our keywords. To mitigate this threat, we removed our keywords before training the model. This helped the

model to learn from the context and not from specific keywords (described in Section 2.3.1). Second, review annotation by crowd workers could yield incorrectly rated reviews, because of their inability to understand the problem well. Thus, two authors annotate the whole training data. Third, the ground truth (of rogue apps) could be biased if it was formed only using top alarming reviews. We mitigated this bias by scrutinizing reviews containing our keywords, which can contain evidence missed by alarming reviews

Threats Remaining

Now, we describe the threats that still remain in our work. First, we investigate only a few thousand apps, which may not be representative of all apps on the Apple App Store[25]. The performance of iROGUE may vary while testing it on all apps of the Apple App Store. Second, we target apps and their reviews only on Apple's App Store. Upon deployment on other app stores, the performance of our approach can differ. Third, if an app distribution platform does not have a similarity recommendation, iROGUE may have to be applied on all apps—a computationally expensive task. However, in such cases, iROGUE can be prioritized for the apps that are flagged by app users (victims). Fourth, some negative reviews (about rogue behavior) may be written by the app's competitors. Identifying such fake reviews is out of the scope of our study.

2.6.2 Limitations and Future Directions

We identify following limitations of this work. Each limitation gives rise to possible future work. First, iROGUE may miss some rogue apps if they do not have alarming reviews at the time of analysis, possibly because they are new apps. However, such rogue apps can be identified as soon as alarming reviews begin to arrive. By leveraging the current evidence, iROGUE helps protect future users and third parties. A possible extension for iROGUE would be to include other information sources such as privacy policies, to identify rogue apps ahead of time.

Second, uncovering rogue functionalities involves manual effort of inspecting top alarming reviews. A possible future direction is to automate this process.

[25] `https://www.apple.com/app-store/`

Example 7: Rogue App from Seed Dataset

App: OurPact Jr. Child App[a]

Rogue Score: 2.47/4

Alarming Review 1

Alarmingness: 4.00/4.00

Date of Review: 2018-06-19

"…however this app shuts down almost everything and can see every text and website you've visited. now, i haven't done anything bad online (recently), but i find that a little creepy and honestly an invasion of privacy. no wonder this app has such crappy reviews. also, i used to have way more apps than i do now. because my parents now have the ability to restrict apps that may be "inappropriate". i already have to ask permission to download apps, so if they were inappropriate my parents wouldn't let me download them. there's too many apps like this and i think kids need a break from all this crap on their devices."

Alarming Review 2

Alarmingness: 4.00/4.00

Date of Review: 2018-08-16

"This is a useless app that no parent need to install I pray for every child who has this app installed on their electronics some parents don't understand the modern society but that's okay (but not really) I'm only given 2 hours and writing this review is using up time WHICH IS NOT FRIKEN OK!!! I hate this hate this app and I hope every child that has had their device attacked by this installment hates this app as much as me. This app should never be okay to use its inappropriate and everybody's children who have this app installed are making there children ANTI-SOCIAL AND VERY NOT COOL. I have many reasons why this app is SOOOOOOO scaring and dreadful so if your reading and thinking about installing this on ur child's device DONT INSTALL IT because that will ruin their future."

[a] `https://apps.apple.com/us/app/id1127917970`

Example 8: Rogue from Snowball Dataset

App: Smart Family Companion[a]

Rogue Score: 2.35/4.00

Alarming Review 1

Alarmingness: 4.00/4.00

Date of Review: 2020-03-15

"How is this even ethical? To put out an app in which you can completely control what's going on on someone else's phone? It's a huge privacy concern. To be honest, apps like this shouldn't exist. It's one thing to put control on a YOUNG CHILD'S phone (which can be done in settings easily) put to put this on an older kids phone is going to destroy trust. No parent should be able to see what their child is doing on their phone 24/7. It's borderline abusive."

Alarming Review 2

Alarmingness: 4.00/4.00

Date of Review: 2019-09-03

"... This app tracks every last thing your child does on their phone. As you can imagine, no 16 year old wants to have their own private life constantly exposed to you. Just because you are their parent, and you live together, doesn't mean that they have to share everything with you. Location, data usage, browsing history, etc. frankly aren't your business. That's their own private information that you don't need to know. Coming from a family with control issues, there is no better way to destroy your relationship with your children. I doubt anyone would want to be around someone who is constantly monitoring and controlling them ..."

[a] `https://apps.apple.com/us/app/smart-family-companion/id1352914754`

Example 9: Rogue App from Snowball Dataset

App: Bark - Parental Controls[a]
Rogue Score: 2.38/4.00

Alarming Review 1
Alarmingness: 4.00/4.00
Date of Review: 2020-05-28
"This app is unfair and invasion of privacy! Kids shouldn't be watched like this all kids cuss and this app tracks that and then snitches on you for it. I don't understand why someone can have so much doubt in their kids. Yes, I understand some kids do some really bad things that shouldn't be done, but if u raise your kids right and teach them right from wrong then you'd be able to trust them. One of my best friends has this app and she literally tells me how much she hates her parents. My friend has never even done anything and she has no reason for this app to be on her phone. I know the internet is dangerous but telling your kids it's dangerous honestly has a bigger effect. Maybe try other methods until it gets to the point of this app abolishing all their freedom and happiness."

Alarming Review 2
Alarmingness: 4.00/4.00
Date of Review: 2021-03-12
"If I could give this zero stars I would. This app is a total invasion of privacy and if you want to ruin your chances of having a relationship with your child, then get this. But if you are one of those parents who don't give your child/teen privacy you are not only hurting them but you are also hurting that bond and relationship with them. As a teen we don't want privacy because we are trying to hide something we just want privacy to be able to feel like our own person . . . "

[a] `https://apps.apple.com/us/app/id1477619146`

CHAPTER

3

METHREE: IDENTIFYING INCIDENTS, EFFECTS, AND REQUESTED ADVICE FROM METOO POSTS

Warning: This chapter may contain trigger words for some readers, especially survivors of sexual harassment.

Survivors of sexual harassment frequently share their experiences on social media, revealing their feelings and emotions and seeking advice. We observed that on Reddit, survivors regularly share long posts that describe a combination of (i) a sexual harassment incident, (ii) its effect on the survivor, including their feelings and emotions, and (iii) the advice being sought. We term such posts MeToo posts, even though they may not be so tagged and may appear in diverse subreddits. A prospective helper (such as a counselor or even a casual reader) must understand a survivor's needs from such posts. But long posts can be time-consuming to read and respond to.

Accordingly, we address the problem of identifying key information from a long MeToo post. We develop a natural language based model to identify sentences from a post that describe any of the above three categories. On ten-fold cross-validation of a dataset, our model achieves a macro F1 score of 0.82.

In addition, we contribute METHREE, a dataset comprising 8,947 labeled sentences identified from Reddit posts.

3.1 Introduction

In the United States, 81% of women and 43% of men have reported some form of sexual harassment or assault in their lifetime.[1] In 2006, Tarana Burke, an activist, coined the *MeToo* phrase for survivors to share their experiences of sexual harassment. This led to what's known as the MeToo movement, which seeks to report sexual harassment and help survivors know they are not alone. Reddit is a popular social media platform that hosts multiple forums called subreddits (r/meToo[2], r/SexualHarassment[3], and r/sexualassault[4]) for survivors to share their MeToo posts.

Prior studies on MeToo posts (Karlekar and Bansal 2018; Hassan et al. 2020; Khatua et al. 2018; Ghosh Chowdhury et al. 2019a) focus on classification. For instance, Ghosh Chowdhury et al. (2019a) identify posts describing MeToo personal stories, Karlekar and Bansal (2018) identify the type of sexual harassment, and Hassan et al. (2020) detect the type of sexual violence. All these existing studies identify relevant MeToo posts from a massive stream of social media text. The expectation is that a prospective helper (e.g., the concerned authority) can provide support to the survivor of identified post. However, merely identifying relevant posts is not enough. A prospective helper must understand (i) what happened, (ii) how sexual harassment has affected the survivor, including the feelings and emotions they are going through (Field-Springer et al. 2021), and (iii) the advice that the survivor is seeking (Andalibi et al. 2016).

Reddit allows a higher number of characters (40k per post) than platforms such as Twitter (250 per post). The MeToo-related subreddits too see long posts (with mean and maximum of 1,881 and 33,432 characters, respectively). For a prospective helper (e.g., the concerned authority), reading long posts that regularly appear on multiple subreddits (O'Neill 2018) can be demanding and time consuming. To not miss out on important parts of long posts, we built a natural language model that identifies (from a MeToo post) sentences describing a sexual harassment incident, its effects on the survivor, and the advice requested. Understanding these important parts can help the prospective helper to address survivors' concerns in responses.

We describe these three sentence categories as follows:

[1] https://www.nsvrc.org/statistics
[2] https://www.reddit.com/r/meToo/
[3] https://www.reddit.com/r/SexualHarassment/
[4] https://www.reddit.com/r/sexualassault/

Sexual harassment incident: Sentences describing unwelcome sexual advances, sexual behavior, requests for sexual favors, verbal or physical acts of sexual nature, offensive jokes or remarks that are either sexual or based on someone's gender.[5]

Effects on the survivor: Survivors describe how they are affected by revealing their feelings and emotions that arise during or after the harassment incident. Examples of effects include the survivor feeling uncomfortable due to the abuser's actions, or being angry or upset due to the harassment.

Requested advice: Sentences in which survivors seek advice from other platform users. Some examples of advice include asking if the survivor's experience is harassment, how to pursue a legal case, and how to confront the abuser.

Example 10: Incident, effects, and request for advice

Categories: Sexual harassment incident, Effects, Requested advice

At *<job-location>*, I was appointed as *<job-title>* a month ago. In my office, this one *<person>* pats my shoulder and I feel his hand has lingered a little too long a couple times. Because of my big history around sexual harassment, I feel extremely uncomfortable with his behavior. I keep thinking if I am considering his behavior inappropriate because of my history? I understand that he wants to be friendly and build rapport, but my body thinks his behavior is little off. Reddit, am I overthinking?

identified sentences

- *In my office, this one <person> pats my shoulder and I feel his hand has lingered a little too long a couple times.*
- *Because of my big history around sexual harassment, I feel extremely uncomfortable with his behavior.*
- *I keep thinking if I am considering his behavior inappropriate because of my history?*
- *Reddit, am I overthinking?*

[5] `https://www.eeoc.gov/sexual-harassment`

Example 10 shows a MeToo post[6] and the three categories of sentences that we identify from it.[7] The identified text describes inappropriate touching and the survivor's uncomfortable feeling. Moreover, it reveals that the survivor is confused and asks if they are overthinking the incident.

In other cases, survivors may ask for advice such as how to report harassment, how to deal with trauma, and so on. Prior research (Field-Springer et al. 2021; Andalibi et al. 2016) shows that it's important to understand and address the effects and the requested advice.

3.1.1 Research Questions

Accordingly, we address the following research questions.

$RQ_{\text{metoo-identify}}$: How can we identify sentences describing the harassment incident, its effects on the survivor, and the requested advice from a MeToo post?

$RQ_{\text{metoo-identify}}$ is important because automatically identifying the three categories of sentences will help a prospective helper understand the incident, the effects on the survivor, and the requested advice without having to read the whole post. Understanding these sentences can helpful prospective helpers in constructing a helpful response.

3.1.2 Contributions and Novelty

We make the following contributions.

- To address both questions, we curate METHREE, a dataset containing 8,947 sentences, labeled for the three categories. Constructing a sufficiently natural and precise dataset turns out to be nontrivial. We leverage active learning for labeling with tractable manual effort.
- To address $RQ_{\text{metoo-identify}}$, we train a natural language model to identify these three categories of sentences from long MeToo posts. Our approach incorporates modern Natural Language Processing (NLP) techniques to achieve strong results.

To the best of our knowledge, we are the first ones to study sentence level identification from long MeToo posts.

[6] Due to space limitations, we have shown a short MeToo post.
[7] The extremely personal MeToo post is paraphrased so that it's not identifiable or searchable.

3.1.3 Key Findings

Our model for identifying three categories of sentences yields a macro F1 score of 82%.

A small qualitative study provides additional validation for our contributions (Section 3.3). For 17 of 20 randomly selected MeToo posts, the identified text is coherent to understand incident, effects, and requested advice. For 16 of 17 posts, we can construct a helpful response without missing out on any crucial information about the survivor's situation.

3.2 The MeThree Dataset and Classifier

We consider the problem of identifying three categories of sentences as multilabel classification task. From a post, the sentences predicted as any of the three categories are identified.

In our adopted active learning approach, the preparation of the dataset and the development of a classifier happen hand-in-hand. For classification, we follow a pool-based active learning approach which is known for training robust models while reducing effort on manual labeling (Hanneke 2014). We curate MeThree, a dataset comprising 8,947 labeled sentences (from subreddits: r/meToo, r/sexualassault, and r/SexualHarassment), and train an XLNet model on MeThree.

Pool-based active learning (Settles 2012) starts with an initial dataset (denoted by L) that we curated by selecting and labeling sentences, most of which contain certain keywords (Section 3.2.1). After curating L, in the active learning process, four steps shown in Figure 3.1 are followed and repeated multiple times. First, a model (denoted by M) is trained on the set L. To do so, we compared the performance of multiple models on the curated L and chose the best-performing one as model M (Section 3.2.2). Second, an unlabeled dataset U is labeled by the predictions of the trained model M. In our case, since most of the sentences in L contained certain keywords, to avoid bias, we selected U from sentences without those keywords and labeled U through M's predictions. Third, from U, data points whose risk of being mispredicted is sufficiently high are queried (using a query method) and labeled manually. Fourth, U is added to L. For the last two steps, we queried misclassified sentences from the set U and labeled them manually (Section 3.2.3). We also selected an appropriate query method for our approach (Section 3.2.4). We repeated the active learning cycle five times to curate the final dataset of 8,947 labeled sentences. We called this dataset as MeThree. In the end, we trained the final model on MeThree to identify sentences (from a long post) that are classified as the incident, its effects, and the requested advice.

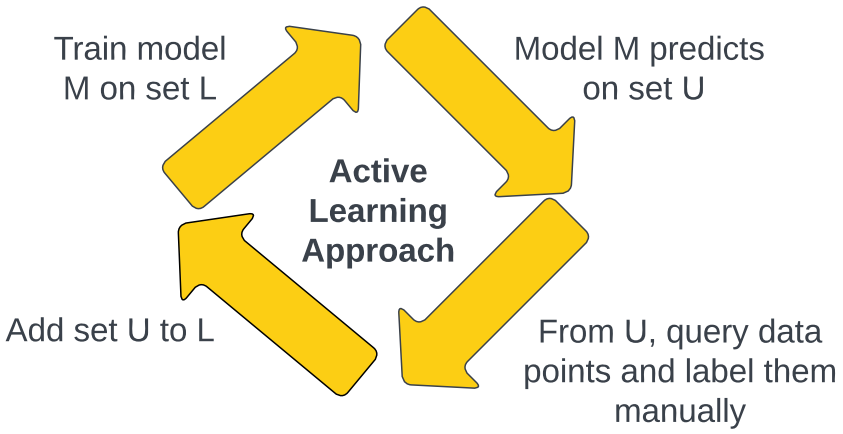


Figure 3.1: Active learning involves four iterative steps.

3.2.1 Initial Training Data for Active Learning

To curate our initial training data (set L) for the active learning approach, we followed four steps. First, we scraped MeToo posts from subreddits. Second, we filtered relevant posts from them. Third, we found candidate sentences for each category. Fourth, we labeled a sample of candidate sentences along with other sentences.

Collecting MeToo Posts

We scraped MeToo posts from three subreddits: r/meToo, r/sexualassault, and r/SexualHarassment, for the period 2016-01-01 to 2021-07-18, using Reddit's Pushshift API.[8] In this process, we collected a total of 9,140 posts. Of these 9,140 posts, there were 263 posts whose content was deleted by the time of our scraping. That's how we were left with 8,877 MeToo posts.

Filtering Relevant MeToo Posts

Some MeToo posts don't share survivors' experiences but share news articles, seek opinions about allegations against celebrities, or promote other platforms. Such posts are irrelevant to our study. We applied the following heuristics to focus on posts containing survivors' personal experiences:

- First-person pronouns: Many survivors while describing their personal experiences, use first-person pronouns in the title of the post. For example, "**I** started to do something about **my** past assault, but instead of feeling better, it actually gets worse" and "**My**

[8] `https://psaw.readthedocs.io/en/latest/`

mom's boyfriend tried to get **me** to do things to him". Thus, we checked the presence of first-person pronouns: *i*, *me*, *my*, and *mine* in the title to identify relevant MeToo posts.

- Advice-related keywords: We observed that survivors also use advice-related keywords in the title. For example, "Need **advice**, or support" and "pls someone read this and **help** me figure out if i was assaulted or not". We used the keyword, *advice*, as seed and queried its synonyms from the Oxford dictionary. We obtained 25 synonyms and manually filtered four of them based on their relevance to our problem. The final list contained *help*, *suggestion*, *advice*, *guide*, and *counsel*. We referred to this list of keywords as *advice keywords*. To filter relevant posts, we checked the presence of these keywords in the title. For identifying synonyms, we also explored other corpora such as WordNet (Miller 1995) but did not find synonyms that were commonly used.

- Advice-related questions: We observed that many relevant posts ask a question (related to sexual harassment) in the title. For example, "Was this rape?" and "Is this sexual harassment?". Such questions are seeking advice without mentioning any of the advice keywords. Using Part-Of-Speech (POS) tagging (Manning 2011), the titles that have an interrogation form and include *rape*, *harassment*, *assault*, and *abuse* as the object, were filtered.

Posts with titles satisfying one of the above rules were filtered out. We checked random 50 filtered posts for relevancy. Of 50 posts, 47 (94%) were relevant because they either sought support or advice related to their case of sexual harassment. Among these 47 posts, we also found one post written by the survivor's friend but the post still expressed the effects on the survivor and sought advice.

In total, we obtained 4,933 relevant posts using the above heuristics. We might have missed some relevant posts, but the objective here is to filter posts with high precision. This is because high precision (94% in our case) means we can build a dataset of relevant sentences without further pruning. Similar heuristics are used in other studies too (Hassan et al. 2020). In this study, out of 4,933 filtered posts, 74.29% (3,665) are from r/sexualassault, followed by r/meToo (17.23%; 850) and r/SexualHarassment (8.47%; 418).

Finding Candidate Sentences

We split each of the 4,933 relevant posts into sentences, using sentence tokenizer of Natural Language Toolkit (NLTK) library[9]. That's how we obtained 102,204 sentences. However, a random sample of these sentences was inefficient in getting sentences that describe incidents,

[9] `https://www.nltk.org/api/nltk.tokenize.html`

or its effects, or requested advice. Thus, we first found candidate sentences of each category using following keywords:

- Sexual harassment incident: Hassan et al. (2020) create a list of 27 MeToo-related verbs (such as *molest*, *touch*, *rape*, *masturbate*). We expanded the list by querying synonyms of these verbs through the Oxford dictionary. The resulting list contained 652 verbs. We manually checked them and found 539 relevant verbs, of which only 313 were unique. We called this final set of 313 verbs as *harassment keywords*. We identified candidate sentences by the presence of one or more harassment keywords in them. That's how we found 30,927 candidate sentences for the incident category.

- Effects on the survivor: For identifying candidate sentences in this category, we leveraged two types of keywords. First, we leveraged the NRC word emotion lexicon (Mohammad and Turney 2013, 2010), which contained a list of words associated with eight emotions. We considered four emotions: *anger*, *disgust*, *fear*, and *sadness*, that a survivor can express, and use lexicons associated with them. In this process, we found 37,271 emotional candidate sentences. Second, we leveraged synonyms of the word, *feel*, that are identified from the Oxford dictionary. We identified 14 synonyms, out of which, eight were relevant to the survivor's feelings. We referred to the final set of keywords (*feel*, *perceive*, *sense*, *experience*, *undergo*, *bear*, *endure*, *suffer*) as *feel keywords*. We found 8,617 candidate sentences containing one or more feel keywords.

- Requested advice: We observed that many questions in MeToo posts are advice seeking. For example, "Was it actually just a mistake and should I forgive him?" and "Am I blowing it out of proportion?". Hence, we considered all questions as candidates for advice seeking sentences. We found 6,354 such candidates. Moreover, we leveraged advice keywords to find an additional 2,678 candidates.

We identified synonyms from the Oxford dictionary. To find such synonyms, we tried corpora such as WordNet (Miller 1995) and PyDictionary[10] but did not find many keywords. For example, while creating the feel keywords, PyDictionary produced no synonyms, and WordNet produced one word, *palpate*, which was uncommon to describe feelings. Thus, we leveraged the Oxford dictionary to identify relevant and commonly used keywords.

Labeling Sentences

Due to presence of keywords (such as harassment keywords, feel keywords, and so on), the candidate sentences are likely to be relevant to the three categories. However, only including

[10] `https://pypi.org/project/PyDictionary/`

candidate sentences can make the training set (set L) biased toward the chosen keywords. Thus, for labeling at this step, we included random 500 sentences not having any keywords, along with 6,900 sampled candidate sentences (including sentences from all sources: harassment keywords, feel keywords, and so on). After discarding duplicates, we were left with 5,947 sentences.

Since a majority of 5,947 sentences still contained chosen keywords, labeling them could still form a biased dataset. Note that this was only the initial training data (set L) in the active learning approach. Later, to mitigate bias, we kept including sentences without any keywords (set U) through multiple repetitions of active learning cycle, as described in Section 3.2.3.

For 5,947 sentences, three of the authors were the annotators. Before labeling, they were aware of the uncomfortable and disturbing text present in these sentences. For each sentence, the annotators were asked the following questions:

1. Does this sentence describe a sexual harassment incident?

2. Does this sentence describe the effects of the incident on the survivor?

3. Does this sentence ask for any advice?

The annotators read each sentence and answered the above questions as either yes or no. Initially, two annotators labeled 200 sentences as per their understanding of the problem statement. Later, they discussed their disagreements and defined the final labeling instructions for all the annotators to follow. The final labeling instructions are described below:

1. Sexual harassment incident: We followed the definition given by the United States Equal Employment Opportunity Commission (EEOC).[11] Any unwelcome sexual advances, sexual behavior, requests for sexual favors, verbal or physical acts of sexual nature, offensive jokes, or remarks that were either sexual or based on someone's gender were labeled as sexual harassment. Sexual harassment is not limited to, and we considered harassment cases with all genders.

2. Effects on the survivor: We considered survivors' feelings and emotions that arose during or after the incident. Examples range from feeling uncomfortable (due to the abuser's actions) to being afraid (emotion: fear) of reporting sexual harassment.

3. Requested advice: We considered sentences in which survivors asked for suggestions on topics related to harassment, e.g., whether to report the incident, where to get therapy from, and how to face the abuser again.

Table 3.1: Relevant examples according to labeling instructions.

Sentence	Incident	Effects	Requested advice
. . . he slid his hand up my leg and into my shorts.	✓		
. . . I was sexually used by <abuser> on many occasions . . . I am in a constant battle with major depression, crippling real event OCD (I ruminate for 16 hours/day) & debilitating anxiety.	✓	✓	
. . . I'm freaking out and have no one to talk to because no one knows about him or what happened . . . What do I do?		✓	✓
Does anyone know how a legal advocate works and what you experienced with them?			✓

Table 3.1 illustrates examples of each category.[12] The first example describes inappropriate physical behavior and is considered sexual harassment. The second example describes that the survivor is sexually exploited (sexual harassment) and suffers from depression and anxiety (effects). In the third example, the survivor expresses fear (by mentioning "freak out") and seeks advice about dealing with it. In the last example, the survivor seeks advice relating to the legal process.

All 5,947 sentences were divided among the three annotators (let's denote them by A_1, A_2, and A_3) such that two of the annotators labeled each sentence. After labeling all the sentences, we obtained Cohen's kappa scores (Cohen 1960) of 0.772 (for sexual harassment incident), 0.774 (for effects), and 0.865 (for requested advice). These scores indicated that we achieved substantial agreement for two categories: sexual harassment incident and effects, and almost perfect agreement for the requested advice category. Table 3.2 also shows Cohen's kappa scores for each pair of annotators. Moreover, the first author resolved all the disagreements. The labeled 5,947 sentences form the initial training data (set L) for active learning.

3.2.2 Initial Model to identify Sentences

After set L is curated, the next step is to train model M. We consider our problem as a multilabel classification task in which each sentence is an input to the model and the output has three binary labels (one label for each category). We trained and evaluated multiple methods on 5,947 labeled sentences (set L) as described below.

[11] `https://www.eeoc.gov/sexual-harassment`

[12] For anonymity, we have masked abusers' details.

Table 3.2: Cohen's kappa scores for each pair of annotators.

Annotators	Incident	Effects	Requested advice
A_1, A_2	0.798	0.793	0.891
A_2, A_3	0.720	0.725	0.843
A_3, A_1	0.795	0.801	0.861
Total	0.772	0.774	0.865

For each of 5,947 sentences, we computed embeddings such as Sentence-BERT (Reimers and Gurevych 2019), TF-IDF (Cahyani and Patasik 2021), GloVe,[13] (Pennington et al. 2014) Word2Vec[14] (Mikolov et al. 2013), and Universal Sentence Encoder (USE) (Cer et al. 2018). For each embedding, the sentence vector was used as an input to a multilabel classifier. For GloVe and Word2Vec, we averaged word vectors to form the sentence vector. For classification, we tried Logistic Regression (LR) (Dreiseitl and Ohno-Machado 2002), Support Vector Machine (SVM) (Cervantes et al. 2020), and Random Forest (RF), and report the best method.

In addition to embedding-based methods, we also applied transformer-based approaches such as RoBERTa (Liu et al. 2019b) and XLNet (Yang et al. 2019). We fine-tuned RoBERTa and XLNet on set L by adding an output layer in the forward direction. The output layer contained three units, one dedicated to each category. Both the models minimized binary cross entropy over five epochs. Moreover, the training batch size and tokenizer length were set to 32 and 256, respectively.

In Table 3.3, we report average F1, precision, and recall scores for the approaches described above over ten-folds of set L. In the same table, we also include the results for searching sentences by category-wise keywords (keywords used in Section 3.2.1). The sentences containing keywords were predicted 1 for that category and others were predicted 0. Moreover, for embeddings-based approaches, Table 3.3 reports the results with only their best-performing classifiers.

TF-IDF, GloVe, Word2Vec, Keyword search, and USE underperform as compared to other methods.

Sentence-BERT followed by SVM achieves the highest macro precision (0.84). However, it shows lower macro recall (0.66) than RoBERTa (0.84) and XLNet (0.87). Overall, XlNet outperforms all other methods by achieving the highest macro F1 score (0.82). Thus, we choose XLNet as our active learning model (model M).

[13]We used Stanford's GloVe model trained on the Wikipedia dataset, which returns a 100-dimension word vector.
[14]We used Word2Vec trained on the Google News dataset and returns a 300-dimension vector.

Table 3.3: Comparing performance of multiple trained models. Bold value for a metric indicates the highest score among all approaches.

		TF-IDF+ LR	GloVe + RF	Word2Vec + RF	Keyword search	USE + SVM	Sentence-BERT + SVM	RoBERTa	XLNet
Incident	F1 Score	0.64	0.43	0.50	0.47	0.69	0.70	**0.77**	**0.77**
	Recall	0.56	0.31	0.33	0.79	0.63	0.61	0.80	**0.82**
	Precision	0.75	0.71	0.73	0.34	0.77	**0.83**	0.74	0.74
Effects	F1 Score	0.66	0.38	0.39	0.49	0.69	0.70	0.78	**0.80**
	Recall	0.60	0.25	0.26	0.82	0.65	0.61	0.80	**0.86**
	Precision	0.73	0.76	0.79	0.35	0.75	**0.84**	0.77	0.74
Requested advice	F1 Score	0.74	0.63	0.64	0.67	0.74	0.80	**0.89**	**0.89**
	Recall	0.70	0.58	0.59	0.93	0.71	0.75	0.91	**0.94**
	Precision	0.78	0.73	0.73	0.53	0.77	0.86	**0.87**	0.84
Macro	F1 Score	0.68	0.48	0.51	0.54	0.71	0.73	0.81	**0.82**
	Recall	0.62	0.38	0.39	0.85	0.66	0.66	0.84	**0.87**
	Precision	0.75	0.73	0.75	0.41	0.76	**0.84**	0.80	0.77

3.2.3 Completing Active Learning Cycles

After model M is trained, it's time to make predictions on the set U and label it. To mitigate the risk of a biased dataset, we chose the set U to be a random sample of 500 sentences not containing any keywords. The already trained model M labeled set U through its predictions. From U, we queried potentially misclassified sentences for manual labeling, using a query method described in Section 3.2.4. Further, the first active learning cycle (Figure 3.1) was completed by adding labeled U to L. We repeated this for four more cycles that involves the training XLNet on the new L, predicting on new U, labeling new U (through M's predictions and manually labeling the queried sentences), and adding new U to L. Overall, a total of five cycles added total 2500 labeled sentences (each time U having 500 sentences without keywords) to the initially 5,947 labeled ones. As a result, the final L became to be of size 8,447. Moreover, while selecting appropriate query method for our approach, as discussed in Section 3.2.4, we labeled additional 500 sentences without keywords. By including all these labeled sentences, we formed the final dataset, METHREE, of size 8,947.

In METHREE, there are 4,331 (48.4%) sentences that belong to at least one category, and 4,616 (51.6%) others. Figure 3.2 shows the Venn distribution of 4,331 sentences among three categories.

Finally, we trained XLNet on MeThree which is used to identify sentences from long posts. Over ten cross validation of MeThree, the model achieved 0.82 macro F1 score (0.78 for incident, 0.79 for effects, and 0.89 for requested advice), 0.86 macro recall (0.82 for incident, 0.83 for effects, and 0.92 for requested advice), and 0.78 macro precision (0.74 for incident, 0.76 for effects, and 0.85 for requested advice).

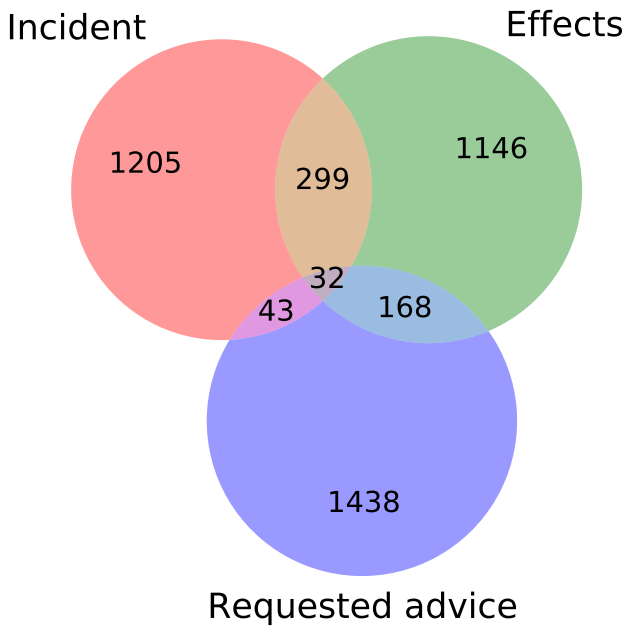


Figure 3.2: Venn diagram showing the distribution of sentences across the three categories.

3.2.4 Selecting Query Method

Uncertainty sampling (Culotta and McCallum 2005; Dagan and Engelson 1995), a widely used querying method, finds uncertain predictions based on the model's prediction probability on the set U. Such uncertain data points are queried for manual labeling. However, uncertainty sampling methods (such as least confidence and entropy) did not work in our case. This is because in the first active learning cycle, model M (XLNet trained on 5,947 sentences; Section 3.2.2) predicted low probabilities on the sentences without any keywords (set U). We

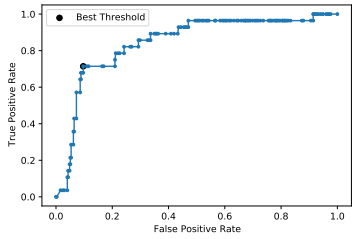


Figure 3.3: ROC curve showing true positive and false positive rates, while considering misclassified incident sentences under positive class. The area under the curve is 0.84. The best threshold is 0.038177.

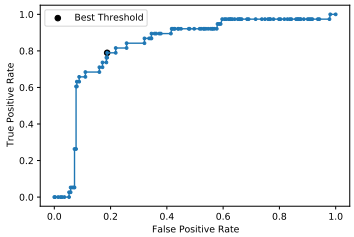


Figure 3.4: ROC curve showing true positive and false positive rates, while considering misclassified effects sentences under positive class. The area under the curve is 0.83. The best threshold is 0.008476.

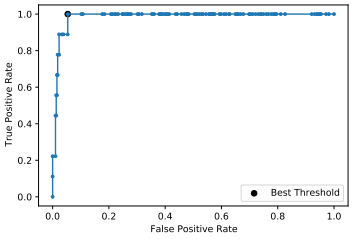


Figure 3.5: ROC curve showing true positive and false positive rates, while considering misclassified requested-advice sentences under positive class. The area under the curve is 0.98. The best threshold is 0.007874.

validated this by predicting on 100 such sentences, where the mean prediction probability was 0.08 (deviation= 0.23) for incident category, 0.10 (deviation= 0.27) for effects, and 0.05 (deviation= 0.21) for requested advice. Due to most of the probabilities being low, uncertainty sampling methods (such as least confidence and entropy) could not discriminate between misclassified and other sentences.

To select an appropriate query method, we found a threshold on the prediction probability, using which we could query from U. To find that threshold, we used a set U', another random sample of 500 sentences not having any keywords. On U', we plotted the Receiver Operating Characteristic (ROC) curve and computed Youden's J-statistic (Youden 1950) as described below. Since this query method was selected during the first active learning cycle, the model M referred below is XLNet trained on 5,947 sentences (Section 3.2.2).

1. The first author labeled the set U'. On 5,947 initially labeled sentences (Section 3.2.1), we achieved substantial agreement for the incident and effects category and almost perfect agreement for the requested advice. Hence, we assumed that all the annotators (three authors) stick to the same labeling definitions and used one of the annotators (the first author) for this small task.

2. The model M made predictions on the set U'.

3. We split the set U' into a set of 400 sentences (set V) and another set containing remaining 100 sentences (set T).

4. We leveraged the set V to fine-tune the threshold. For each category, we considered the

misclassified sentences in set V and found a threshold (on the predictions' probability) that could retrieve them. We found that out of 400 sentences, model M misclassified 28 sentences for the incident category, 38 for effects, and 9 for the requested advice category. Since we needed to retrieve these sentences, for each category, we considered misclassified sentences under positive class and others under negative class, while plotting true positive and false positive rates in ROC. Figures 3.3- 3.5 show ROC and the area under the curve for each category. From each ROC curve, we found the best threshold (using Youden's J-statistic (Youden 1950)) that maximized recall (for positive class) and minimized false positive rate.

For the incident category, we found 0.038177 as the threshold, above or equal to which sentences can be queried. Similarly, we found a threshold of 0.008476 for the effects category and 0.007874 for the requested advice category. We also tried combining these three thresholds into a single threshold but that did not query more misclassified sentences than the individual threshold case.

5. For each category, to ensure that we did not miss out on the misclassified sentences, we also queried 30 sentences below the threshold for every 100 predictions. For example, the model M predicted on the set V which has 400 sentences (4 times 100), we also queried 4*30 = 120 sentences below the threshold for each category.

   To sum up, our query method is: for each category, query (i) sentences with prediction probability above or equal to the threshold, and (ii) 30 sentences below the threshold for every 100 predictions.

   Using the above query method, we could query the following number of misclassified sentences from V: 25 (89.28%) of 28 for the incident, 35 (92.10%) of 38 for effects, and 9 (100%) of 9 for requested advice. Since set V was used for fine-tuning, we also tested our query method on the unseen set T.

6. We leveraged the set T to test our query method. In the set T, M misclassified 10 sentences in the incident category, 4 in the effects, and 3 in the requested advice. Our query method retrieved 9 (90%) misclassified incident sentences and all misclassified cases (100%) in the other two categories.

For each time the active learning cycle was repeated (discussed in Section 3.2.3), we used the above query method to retrieve potential misclassified sentences from U and manually labeled retrieved sentences. For manual labeling, each of the three annotators (authors) labeled retrieved sentences for a category.

3.3 Qualitative Analysis

We applied the final XLNet model (trained on METHREE as described in Section 3.2.3) on random 20 posts (containing at least a thousand characters) and followed the below steps for each post.

First, we split the post into multiple sentences, using the sentence tokenizer of the NLTK library.[15]. Second, we provided all the sentences as input to model M and arranged the identified sentences in the order they were present in the post. Third, along with the post title, we read the identified sentences in the arranged order and checked if identified sentences are coherent to understand the incident, effects, and requested advice. For 17 out of 20 posts, the identified text was coherent.

Further, we divide 17 posts and their identified text among three annotators (the same authors A_1, A_2, and A_3) such that each post was read by one annotator and its identified text was read by the other. Each annotator was asked to construct a supportive or advice-offering response based on details present in the given text. Providing such responses is one kind of help to the survivor (Schneider and Carpenter 2019; Andalibi et al. 2016). For each post, the first author analyzed the difference between the response to the post (R_p) and the response to the identified text (R_e). Only in 1 of 17 cases, a crucial detail (about the survivor's situation) was missed by R_e that was part of R_p. This was because that detail was also missing from the identified text. However, for 16 of 17 cases, R_e did not miss out any crucial details that were part of R_p. Our model can potentially be used to understand the essential details (without reading long posts) and construct a helpful response based on the identified text. In turn, this can speed up the process of providing help on a large scale.

3.4 Related Work

There has been extensive research in analyzing MeToo posts and finding useful insights (Manikonda et al. 2018; Gautam et al. 2020; Deal et al. 2020; Field et al. 2019; Reyes-Menendez et al. 2020). However, only a few studies have looked MeToo experiences from classification perpective. Karlekar and Bansal (2018) leverage the MeToo experiences posted on the SafeCity website[16], an online forum to report sexual harassment. They collect 9,892 MeToo experiences that convey one of the three types of harassment: (i) groping or touching, (ii) staring or ogling, and (iii) commenting. Further, they train a deep neural network to identify the type of harassment experienced by the survivor. Yan et al. (2019) improve the performance of this clas-

[15] `https://www.nltk.org/api/nltk.tokenize.html`
[16] `https://www.safecity.in/`

sification by proposing a quantum-inspired density matrix encoder. Liu et al. (2019a) leverage the same dataset and annotate it for attributes such as the abuser's age (below 30 or older), the abuser's relation with the survivor (for example, relative or teacher), location of harassment (for example, park or street). They propose a framework to identify these attributes from a MeToo experience. Bauer et al. (2020) also leverage the SafeCity dataset and build a chatbot system to help survivors. The SafeCity dataset contains concise experiences (typically 3-4 sentences long) and is unfit to identify sentences in our case.

Moreover, Hassan et al. (2020) train a model on 520,761 #MeToo hashtag tweets to identify tweet level attributes, such as the category of sexual violence reported, the survivor's identity (tweeter or not), the survivor's gender. They also achieve 80.4% precision and 83.4% recall in identifying sexual violence reports. Ghosh Chowdhury et al. (2019b) label 5,119 tweets for types: (i) disclosure and (ii) nondisclosure. The tweets that include a survivor's personal experience are annotated as disclosure and others as non-disclosure. Out of 5,119, they find 1,126 (22%) tweets under disclosure category. Moreover, they propose a language model to classify tweets into two types. Moreover, other studies such as Khatua et al. (2018) and Ghosh Chowdhury et al. (2019a) also focus on similar classification tasks. All these works perform classification tasks on each MeToo post. However, our work focuses on sentence-level identification of sexual harassment incident, its effects on the survivor, and requested advice.

Our work is the first attempt to identify text from long MeToo posts to the best of our knowledge.

3.5 Discussion

We now discuss our conclusion, our work's limitations, and possible future directions.

3.5.1 Conclusion

The survivors of sexual harassment frequently share their long MeToo posts on subreddits. Using the active learning approach, we trained XLNet model to identify sentences describing (i) the sexual harassment incident, (ii) the effects on the survivor, and (iii) the requested advice, from such posts. We also curated MeThree, a dataset of 8,947 sentences labeled for the three categories. On ten-fold cross-validation of MeThree, our model achieved a macro F1 score of 0.82. The sentences identified by our model can help a prospective helper understand essential details without having to read the entire post. As a result, it can potentially speed up the process of providing help to the survivors.

3.5.2 Limitations and Future Work

Our work suffers from some limitations, and a few of them also motivate future directions of improvement. First, sometimes the identified sentences may not be coherent or miss some details about the survivor's situation. That's why we don't claim our model to be a summarization tool. However, according to our analysis in Section 3.3, non coherent cases and the cases requiring details beyond the identified text are only a few (4 of 20). In the future, our work could be extended to identify other important sentences which can summarize the whole post. Second, our model is trained on the sentences scraped from only three subreddits. We expect the nature of sentences in MeThree to be similar to sentences present on other MeToo-related subreddits. However, we plan to fine-tune the model before applying it on the other subreddits.

We can also extend our work to generate an automated response based on the identified incidents, effects, and requested advice. The automated response after slight corrections by human interventions can offer support and advice to the survivor. Moreover, the similarity between the advice-seeking sentences and users' responses can assess how relevant each response is. This way platform will be able to show highly relevant responses above the less helpful ones.

3.6 Broader Perspective and Ethical Considerations

Although we identified text from long posts to help survivors, we acknowledge some limitations and possible misinterpretations that may occur, especially with the data on such a sensitive topic. We discuss them below.

1. **Consent:** Our data was scraped from the public Reddit posts. Hence, we did not take the consent of the survivors writing such posts. Moreover, as described by Ghosh Chowdhury et al. (2019b), some survivors may get uncomfortable if they are reached out for consent.

2. **Anonymity:** We did not save survivors' personal information such as usernames, or users' history of posts. For the example sentences presented in this paper, we also removed potentially identifying information, such as the survivor's age, job title, and location. Moreover, we paraphrased the example MeToo post. We don't plan to release MeThree publicly.

3. **Labeling disturbing text:** The sentences from MeToo posts can be disturbing to read, especially for people who have gone through a similar experience. Therefore, we didn't hire crowd workers or volunteers to for any labeling task. Instead, the three authors of this paper did it.

4. **Potential misinterpretation:** We were extremely aware of the sensitivity of this research before labeling sentences. However, we may have misinterpreted some MeToo experiences. That's why we don't claim that our labeling is fully accurate.

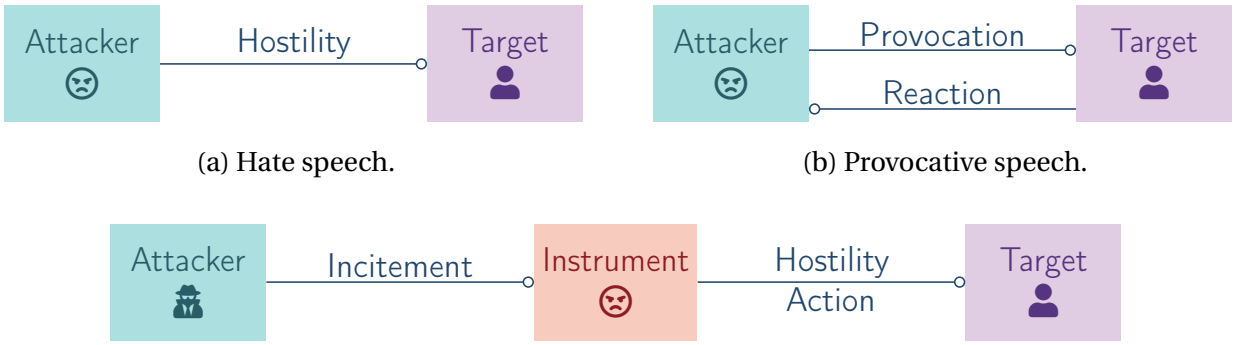
CHAPTER

4

UNDERSTANDING INCITING SPEECH IN SOCIAL MEDIA

Warning: This chapter includes examples of Islamophobic postings on social media.

BACKGROUND: Prior social-media research focuses on identifying hate speech, whereas inciting speech that is subtle and harder to identify, has not been well studied. PROBLEM: On platforms such as WhatsApp, incitement is often used against religious groups to instill anger and violence. We focus on identifying such inciting speech that is targeted against Muslims. DATA AND METHOD: We leverage an existing dataset of Indian WhatsApp posts that are Islamophobic. An analysis of Islamophobic posts revealed three rhetorical strategies of incitement. We randomly selected and labeled 7,000 sentences for three rhetorical forms and trained deep learning–based models to automatically identify such strategies. We further qualitatively analyzed inciting speech to identify impoliteness super-strategies commonly employed. FINDINGS: Our computational method achieved an average AUC score (Area Under the Curve) of 0.851 over five-fold cross-validation. Moreover, Our pragmatic analysis revealed two impoliteness super-strategies used in inciting speech. Further, our computational model identified 19,245 inciting sentences from unseen non-Islamophobic posts. IMPLICATIONS: Automatically identifying inciting speech on social media can prevent such text from being posted. This may reduce online antisocial behavior and potential harm in the real-life scenario



(a) Hate speech.

(b) Provocative speech.

(c) Inciting speech.

Figure 4.1: Three kinds of malice on social media. Hate speech involves an attacker expressing hatred toward a target. Provocative speech involves an attacker provoking a target to elicit a reaction from them (such a reaction may have the target lose face or otherwise advance the attacker's agenda). Inciting speech involves the attacker riling up an intermediary to use as an instrument in expressing hatred or carrying out a malicious action on the target. We focus on inciting speech.

such as violence and discrimination.

4.1 Introduction

We investigate *inciting speech* as a distinct type of antisocial communication on social media. Previous works (Łubiński 2017; Alexander 2001) on inciting speech only focused on violence against a target. However, we follow the broad definition given by International Covenant on Civil and Political Rights (ICCPR)[1]. Basically, we consider inciting speech as one that stimulates its recipient's hostility, or anger, or urges them to take an action such as discrimination or violence against a target.

Interestingly, inciting speech has not been intensively studied on social media and is largely overshadowed by hate and provocative speech (Calvert 1997). We posit that inciting speech is a distinct category in its own right. Figure 4.1 shows how these three categories of malicious speech are structurally different. Both hate (UN-hatedefn) and provocative speech (Boudana and Segev 2017) involve two parties (*attacker* hating or provoking *target*), whereas in incitement the attacker incites the instrument (non-target group of people, who are the recipients of such speech) to elicit anger, hostility, or action against the target.

We begin our study of incitement from an existing dataset of 1,142 Islamophobic posts

[1] `https://www.ohchr.org/sites/default/files/Documents/Issues/Opinion/SeminarRabat/Rabat_threshold_test.pdf`

extracted from public WhatsApp groups that discuss Indian politics (Saha et al. 2021).

Example 11 shows one of these posts illustrating inciting speech. The underlined snippet typecasts Muslims and seeks to engender hostility toward them. Importantly, this post qualifies as incitement even though it doesn't use derogatory words. Inciting speech is thus missed by hate speech detectors that are based on keywords.

Example 11: Inciting speech against Muslims

"Leave chatting and read this post or else all your life will be left in chatting. In 1378, a part was separated from India, became an Islamic nation - named Iran...and now Uttar Pradesh, Assam and Kerala are on the verge of becoming an Islamic state...People who do love jihad is a Muslim. Those who think of ruining the country - Every single one of them is a Muslim !!!! Everyone who does not share this message forward should be a Muslim..."

4.2 Rhetorical Strategies in Incitement

We adopt thematic analysis (Nowell et al. 2017) focused on coding reliability to understand how inciting speech is expressed on social media. To this end, we analyzed more than 100 Islamophobic posts and identified three prevalent rhetorical strategies in such posts. We observed that a post may exhibit multiple strategies but a sentence typically has no more than one. We focus on individual sentences to make our analysis more robust; a post can be considered inciting if it contains at least one inciting sentence. We use a well-known natural language processing toolkit called NLTK (Nltk) to split posts into sentences.

Identity Sentences that criticize or stereotype people identifying with the target group, including their beliefs such as their scriptures, practices, or leaders. Example 12 illustrates this rhetorical strategy, which may be realized through explicit defamation or implicature through disingenuous rhetorical questions.

Imputed misdeeds Sentences highlighting past or ongoing misdeeds such as oppression or violence (either real incidents or fake) of the target group. Example 13 illustrates this rhetorical strategy, which may be realized through accusation or indirectly by presenting "facts".

Exhortation Sentences that urge its readers to act against the target. Such actions include violence, discriminating against people of the target group, or boycotting them. We also

considered subtle cases. Example 14 illustrates this rhetorical strategy, which may be realized through explicit commands or indirect hints for what must be done.

Example 12: Rhetorical strategies: Identity

"…Hindu-Muslim unity is impossible because the Muslim Quran does not tolerate Hinduism as a friend."

"Now you can think that whom will these Muslims serve as IAS …Islam or country?"

"The testimony of Kafir in Sharia court, ie Qazi court, is not valid"

"…Which superstitious religions are in India * * Yes Islam is the anti-human religion of superstition * * 1) If Allah , God is equally powerful * * There is God and he hates Kafiro * * And Kafiro does not have the right to live"

Example 13: Rhetorical strategies: Imputed misdeeds

"…Only Hindu temples are destroyed and business and other activities are being increased in their place and Hindus are also being attacked!"

"Entered, broke the Shivling into pieces, and acquired as much property as he could in the solution"

"Muslim doctor silently sterilized 4000 Buddhist women …"

4.3 Analyzing Incitement Strategies

We now take a closer look at the above three incitement strategies to establish their robustness as constructs.

Example 14: Rhetorical strategies: Exhortation

"Hindu society should stop worshiping these tombstones, tombs, pirs."
"Be violent for religion"

"If the madrasa is not closed, after 15 years, more than half of the Muslims of the country will be supportive of the ISIS ideology"

4.3.1 Generality of the Strategies

The above three rhetorical strategies are generic and apply to a variety of targets. To show their generality, we examined Mathew et al.'s (Mathew et al. 2021) HateXplain2021 dataset, which consists of Twitter and Gab posts along with an indication of whether each post is hate speech and which target is involved in it. We found many inciting posts in HateXplain2021, addressing a variety of targets. Example 15 shows three such posts from that dataset.

Example 15: Incitement in other domains

Targeting whites (Identity, imputed misdeeds)
"<number> year old white hat boys are the worst breed. All they do is recycle their same old holocaust racist sexist and homophobic jokes to hide their insecurities. I got news for you buddy it not working"

Targeting zionists (Identity, exhortation, and imputed misdeeds)
"Zionism is the epitome of evil. We must demand that US congress ceases the funding of the atrocities committed by zionists against Palestinian Gazans Lebanese. Any other Muslim country or people of Iran will never bow to Israel or Saudi Arabia the world will not either."

Targeting non-whites (exhortation)
"Studies show non-whites view freedom completely different than whites, for example, <number> of Hispanics view hate speech as an act of violence which means they would ban it ... their view of hate speech could be anything, wake up your family and friends"

The first post in Example 15 exemplifies two strategies: identity and imputed misdeeds. The second post exemplifies all three strategies. The third post exemplifies the exhortation strategy.

Importantly, the posts in Example 15 do not fall under hate or offensive speech (according

to Mathew et al.'s annotations), meaning that they would be missed by traditional tools.

We curate a dataset, INCITE, containing 7,000 sentences (from Islamophobic posts) that are labeled for three rhetorical strategies. Details of curating INCITE and implementing our computational models are discussed in Section 4.5.

4.3.2 Textual Signatures of Incitement Strategies

We computed embeddings for inciting sentences in INCITE using the Universal Sentence Encoder (USE) (Cer et al. 2018). These embeddings are 512-dimensional vectors. We applied a nonlinear dimensionality reduction approach, called t-distributed Stochastic Neighbor Embedding (t-SNE) (Gisbrecht et al. 2015) to reduce them to two dimensions. We computed the average vector for each category, as shown in Figure 4.2.

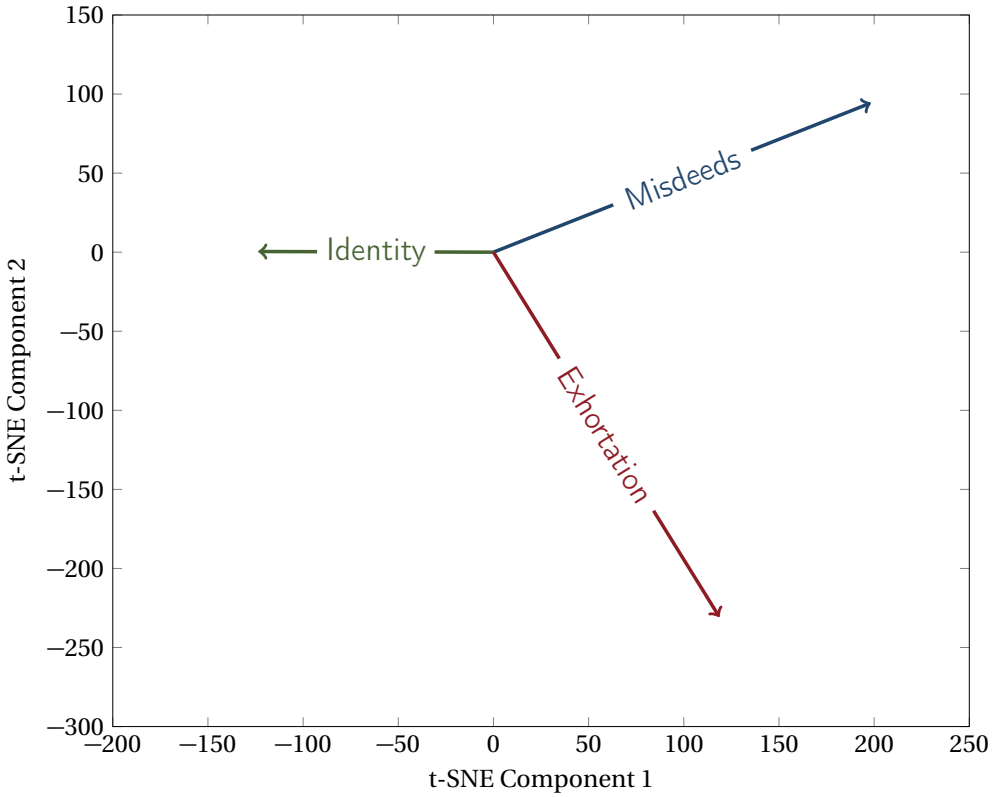
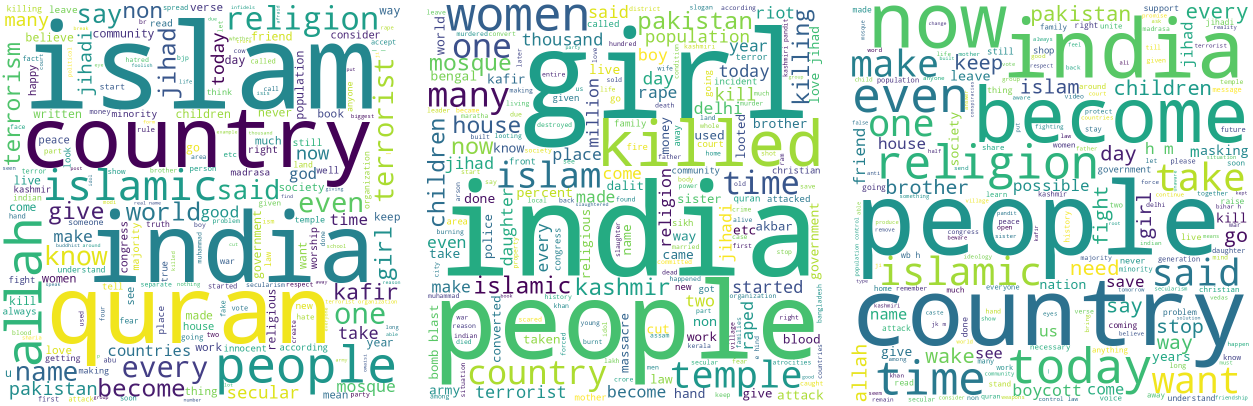


Figure 4.2: Showing the average sentence vector for each rhetorical category after t-SNE dimensionality reduction.

The vectors for identity and imputed misdeeds are almost opposite (cosine similarity of –0.90). The vectors for exhortation and identity are farther apart than orthogonal (cosine

similarity of –0.46). And, the vectors for imputed misdeeds and exhortation vectors are almost orthogonal (cosine similarity of 0.03). The spatial distribution of these three vectors makes clear that the three rhetorical strategies are well-separated from each other.

In INCITE labeled corpus, we analyzed the words commonly used for each rhetorical strategy. Figures 4.3c–4.3b show word clouds containing such words.[2] For the identity strategy, we found that the choice of words such as *allah*, *Islam*, *quran*, and *mosque*, is inclined toward the target community and its beliefs. Moreover, we found that sometimes the text stereotypes the target through words such as *terrorist* and *jihadi*.



(a) Identity. (b) Imputed misdeeds. (c) Exhortation.

Figure 4.3: Prominent words in the three rhetorical strategies.

For imputed misdeeds, we found words related to the victims of oppression or violence such as *girl*, *women*, *daughter*, *kafir*, and *children*. Moreover, there are words describing oppression or violence such as *killed*, *raped*, *cut*, *riot*, *terrorist*, *jihadi*, and *bomb*. For exhortation, words such as *country* and *India* are prominent. This is because such sentences ask readers to eradicate Muslims (or their possessions) because of their negative impact on the 'country' or 'India'. In addition, we observed many action-describing words such as *make*, *raise*, *wake*, *voice*, *stand*, and *fight*. Some words such as *people*, *India* were prominent in all three strategies.

Figure 4.4 shows the distribution of the annotated sentences in INCITE. Out of 7,000 sentences, 433 (6.19%) represent exhortation, 1,065 (15.21%) represent identity strategy, and 1,579 (22.55%) represent imputed misdeeds. Moreover, 3,923 (56.05%) sentences do not belong to any of these strategies (None category).

[2]Two words, *Hindus* and *Muslims* were among the most frequent words. We removed these two words from the text to visualize the remaining frequent words.

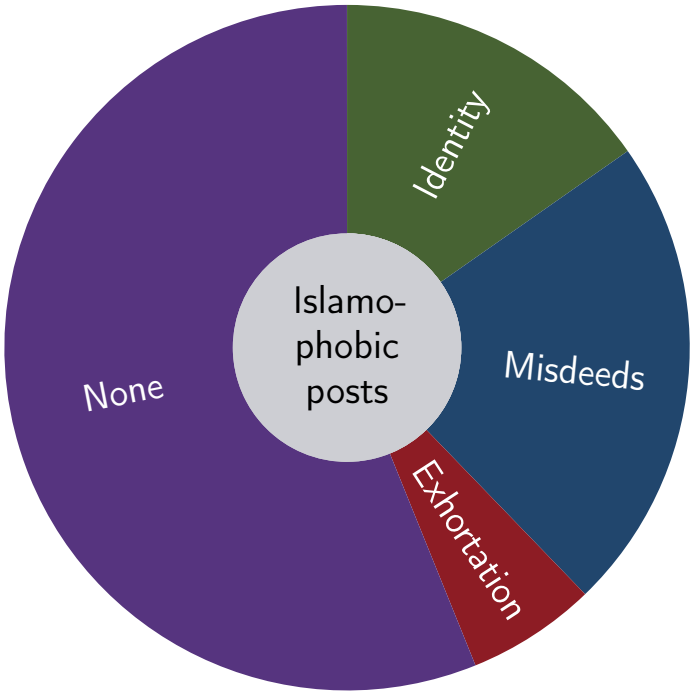


Figure 4.4: Distribution of sentences across the three rhetorical strategies and None.

4.4 Impoliteness Super-Strategies in Incitement

Culpeper (Culpeper 2011) defined impoliteness as a negative attitude toward specific behaviors. We observe that two of the rhetorical strategies (identity and imputed misdeeds) attack a target directly and thus illustrate impolite language. However, exhortation urges action against the target but is not directly about the target; thus, it doesn't illustrate impoliteness. Table 4.1 summarizes five super-strategies for impoliteness due to Culpeper's (Culpeper et al. 2003).

Accordingly, we randomly sampled and analyzed 100 sentences from the identity and imputed misdeeds categories of INCITE.

Two super-strategies, bald on record and negative impoliteness, were prevalent in this sample. For example, *"Muslims are not friends of anyone"* directly attacks the target (Muslims) (bald on record).

For negative impoliteness, we found cases that indirectly associate the target with a negative aspect. For example, *"Alauddin Khilji summoned Rana Ratan Singh of Chittor on the pretext of friendship and then killed [him]"*, associates Muslims (because Alauddin Khilji was a Muslim emperor) with a negative aspect for killing Hindus (because Rana Ratan Singh was a Hindu emperor). This sentence indirectly implies that Muslims always oppress Hindus.

We found no sentences of positive impoliteness. We found only one sentence out of 100

Table 4.1: Culpeper's super-strategies for impoliteness and their prevalence in a sample of 100 Islamophobic sentences.

Super strategy	Description	Prevalence
Bald on record	Direct attack to the face of the target	Prevalent
Positive impoliteness	Attack the positive face of the target by ignoring or excluding them, being unsympathetic and unconcerned toward them, making them uncomfortable, and using obscure and taboo language	None
Negative impoliteness	Attack the negative face of the target by associating the target with a negative aspect, condescending, frightening, or ridiculing them, and invading their space	Prevalent
Sarcasm	State the opposite of the literal meaning to express a negative attitude	Rare
Withhold politeness	Be silent or fail to act where politeness is expected, e.g., by forgetting to say thanks	Not applicable

exhibiting sarcasm, apparently against Muslims claiming minority status in India: "India has the largest Muslim population in the world after Indonesia ???? Oddly enough, it is still a minority ??????".

Since the sentences in our dataset are not threaded or linked with author identities, withhold politeness is ruled out. Moreover, we did not find language showing positive impoliteness.

4.5 Method

Figure 4.5 shows the overview of our method. Our method includes two phases. First, we leveraged Islamophobic posts collected by Saha et al. (Saha et al. 2021) and curated INCITE, a dataset of inciting sentences (Section 4.5.1). Second, we trained and evaluated multiple embeddings-based and transformer-based models over five-fold cross-validation of INCITE (Section 4.5.2). Further, we chose the best-performing model for the identification of inciting sentences.

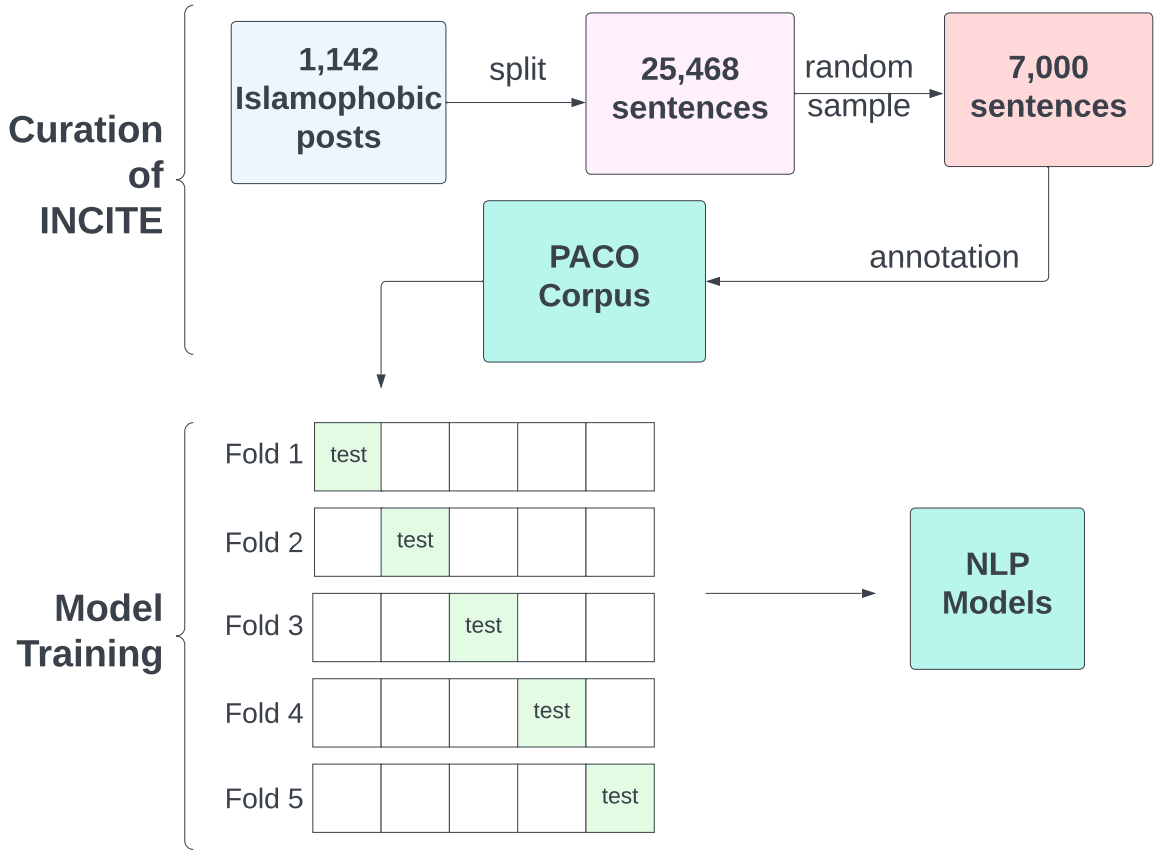


Figure 4.5: Overview of our method. First, we curated a dataset of inciting sentences called INCITE. Second, we leveraged INCITE to train multiple models and choose the best one.

4.5.1 Curation of INCITE

Saha et al. (Saha et al. 2021) identify posts that can instill fear of Muslims, scraped from Indian public WhatsApp groups discussing politics. Out of 27k curated posts, they find ∼8,000 Islamophobic posts. However, they only shared 4,782 posts publicly, out of which 1,142 posts were Islamophobic. As indicated in Example 11, Islamophobic posts also contain inciting sentences. Hence, we split 1,142 Islamophobic posts into 25,468 sentences (using the sentence tokenizer of Natural Language Toolkit (NLTK) library (Nltk)) and randomly sampled 7,000 sentences for annotation purposes.

The first two authors were the annotators. They were given a sentence and asked to label one of the following categories: (i) identity, (ii) imputed misdeeds, (iii) exhortation, and (iv) none. Along with the sentence to annotate, they were provided its preceding and succeeding sentences to get enough context while labeling.

The annotation was conducted in three phases. In the first phase, the annotators were provided with initial labeling instructions, which they used to label a total of 400 sentences, in four rounds of 100 sentences each. In this phase, Cohen's kappa score came out to be 0.48 (moderate agreement) (Cohen 1960). After each round, the annotators discussed their disagreements, which helped in finalizing the labeling instructions. In the second phase, the two annotators labeled 600 sentences using the final instructions, leading to Cohen's kappa score of 0.73 (substantial agreement). In the third phase, the remaining 6,000 sampled sentences were split among the two annotators, such that only one annotator labeled a sentence.

The final labeling instructions, including definitions and examples of each rhetorical strategy, are shown below:

- **Exhortation:** Sentences that urge readers to act against the target were labeled as exhortation. For example, the following sentence asks readers to support Hindus in the fight against Muslims. This support-seeking is open to interpretation and can mean anything from boycotts to violence against Muslims. Thus, we considered such cases in this category.

  "Always support Hindus in the fight of Hindu Muslim, right wrong no matter, all of them later!"

  We also considered subtle cases that indicate some action. For example, an indirect call to shut down a madrasa (Islamic school) is depicted in the following:

  "If the madrasa is not closed, after 15 years, more than half of the Muslims of the country will be supportive of the ISIS ideology"

- **Imputed misdeeds:** Sentences that express oppressing or violent incidents (real or fake) as facts, to provoke readers against a particular religious group were labeled as imputed misdeeds. A relevant sentence structure is "X did violence to Y", where X is Muslim(s) in our use case and Y is an individual or some other group of people. Such an example is presented below.

  "<A person> - set fire - heavy damage at Sartala; 3 out of four rooms were destroyed"

- **Identity:** Sentences that criticize or stereotype the target or target's beliefs such as their sacred books, leaders, were labeled relevant for this strategy. Following are a few examples:

"In Kashmir, every person who speaks 'Murdabad' is a Muslim" (targeting Muslims)

"…Muslim children are well taught Jihadi Quran in madrasa" (targeting Quran, the main Islamic scripture)

The first example above targets all Muslims that they say 'Murdabad' in Kashmir (meaning India's dismissal). Moreover, the second example mentions the Quran to be Jihadi, meaning it spreads terrorism. Indirectly, the latter example typecasts all Muslim children to become terrorists. Such sentences invoke anger against a religious group and hence are considered incitement.

The sentences not indicating any of the above types were labeled 'none'. Combining the sentences annotated in the three phases, we curated a dataset of 7,000 sentences. We call this dataset INCITE.

Ethics note: We annotated the sentences present in Islamophobic posts. Such Islamophobic posts were shared by Saha et al. (2021) and did not include any private information of WhatsApp users who wrote them. Moreover, since these posts were part of public WhatsApp groups, neither Saha et al. (2021) nor us were required to take the consent of WhatsApp users before using the text. We acknowledge that the nature of the text can be disturbing, especially for Muslims. That's why we did not hire crowd workers to annotate sentences. Instead, the authors of this paper who were aware of the nature of the text completed the annotation.

4.5.2 Model Training

We considered our identification problem a multiclass classification task. For multiclass classification, we explored multiple training approaches on INCITE. Each sentence was input to a model and the output was one of the four classes: (i) identity, (ii) imputed misdeeds, (iii) exhortation, and (iv) none. We discuss multiple approaches and compare their performances below.

TF-IDF weighs each word in the corpus according to its Term Frequency (TF) and Inverse Document Frequency (IDF) (Cahyani and Patasik 2021). Based on the number of unique tokens in INCITE, TF-IDF yielded a 11,441-dimensional embedding for each sentence of this dataset. We provided these embeddings as input to multiple classifiers such as Logistic Regression (LR) (Dreiseitl and Ohno-Machado 2002), Random Forest (RF) (Zakariah 2014), and Support Vector Machine (SVM) (Cervantes et al. 2020), and compared their performance.

Word2Vec converted each word in INCITE into a 300-dimensional embedding (Mikolov et al. 2013). In our case, we obtained such word embeddings using the Word2Vec model pre-trained on the Google News dataset. To obtain a sentence embedding, we averaged Word2Vec embeddings for all words present in that sentence. Further, sentence embeddings were provided as inputs to LR, RF, and SVM.

GloVe yields an embedding for each word in the corpus (Pennington et al. 2014). We used Stanford's GloVe model, which is trained on the Wikipedia dataset and returns a 100-dimensional word embedding. We averaged these word embeddings in the same way as we did in Word2Vec. Finally, we trained the above three classifiers on GloVe sentence embeddings.

Universal Sentence Encoder (USE) leverages Deep Averaging Network (DAN) to extract 512-dimensional embeddings for each sentence (Cer et al. 2018). We leveraged these embeddings as features of the above classifiers.

Transformer-based models: We leveraged modern transformer-based approaches such as BERT (Devlin et al. 2019), RoBERTa (Liu et al. 2019b), and XLNet (Yang et al. 2019). We fine-tuned all these models on our dataset by adding a layer (with softmax activation) in the forward direction, containing four output units (one for each class). Further, these models were trained using a batch size of 32, the maximum sequence length of 256, for five epochs to minimize the cross-entropy loss.

All the above approaches were evaluated on INCITE for five-fold cross-validation. Moreover, INCITE was divided into five folds in a stratified manner, leading to the same class distribution in each fold. For evaluation, prior works on hate and fear speech (Saha et al. 2021; Pereira-Kohatsu et al. 2019; Salminen et al. 2020) leveraged the AUC-ROC metric because it measures the goodness of fit, especially appropriate for imbalanced datasets. INCITE also suffers from imbalanced class distribution (Figure 4.4). Hence, we also leveraged AUC-ROC score for evaluation.

Table 4.2 shows AUC-ROC score (obtained by one versus one method (Onevsone)) achieved by each of the above approaches in five folds. Among embeddings-based methods, TF-IDF with LR achieved 0.818 as the average AUC-ROC score, followed by Word2Vec with SVM (0.815), USE with SVM (0.812), and GloVe with SVM (0.800). However, all transformer-based approaches such as BERT (0.832 average AUC-ROC), RoBERTa (0.851 average AUC-ROC), and XLNet (0.845 average AUC-ROC) outperformed embedding-based approaches. Overall, RoBERTa achieved the highest average AUC-ROC score and was chosen as the best model.

Table 4.2: AUC-ROC score for each of the five folds. Bold indicates the highest average AUC-ROC score among all approaches.

Approach	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Average
TF-IDF + LR	0.810	0.825	0.832	0.805	0.820	0.818
TF-IDF + RF	0.790	0.793	0.812	0.786	0.803	0.780
TF-IDF + SVM	0.810	0.818	0.832	0.801	0.812	0.814
Word2Vec + LR	0.810	0.800	0.788	0.792	0.784	0.794
Word2Vec + RF	0.756	0.752	0.747	0.745	0.758	0.751
Word2Vec + SVM	0.823	0.820	0.805	0.816	0.814	0.815
GloVe + LR	0.784	0.785	0.775	0.755	0.766	0.773
GloVe + RF	0.760	0.758	0.752	0.760	0.758	0.757
GloVe + SVM	0.809	0.812	0.793	0.794	0.795	0.800
USE + LR	0.812	0.825	0.801	0.786	0.807	0.806
USE + RF	0.768	0.773	0.760	0.754	0.757	0.762
USE + SVM	0.821	0.829	0.811	0.798	0.803	0.812
BERT	0.838	0.826	0.839	0.819	0.838	0.832
RoBERTa	0.851	0.849	0.854	0.845	0.856	**0.851**
XLNet	0.843	0.850	0.856	0.833	0.847	0.845

4.6 Contributions and Novelty

We identified incitement as a largely under-studied but important form of antisocial communication prevalent on social media. We identified three major rhetorical strategies used for incitement, showing that they are illustrated in a variety of ways. Despite that, these strategies are remarkably well-separated as shown in embedding space. Our computational tool helps us identify these strategies with high reliability.

4.7 Related Work

In this section, we discuss existing studies related to hate, dangerous, and fear speech and show how our work is different from theirs.

4.7.1 Hate Speech

There is no all-encompassing definition of hate speech (Benesch 2014). But typically, hate speech is considered abusive speech or a direct serious attack on an individual or a group based on the attributes such as race, ethnicity, religion, and sexual orientation (Sellars 2016; ElSherief et al. 2018). Due to direct and abusive attacks, hate speech often contains derogatory words such as *n*gger* and *a**hole* that are used against the target. However, inciting speech doesn't necessarily have derogatory words. For example, *"Question- What is non-violence... ?? Answer: Bakra Eid"* uses sarcastic language to incite readers against the Muslim tradition of sacrificing goats ('Bakra' in Hindi) during the Eid-ul-Adha celebration. As a result, this sentence falls under the identity strategy of incitement but not hate speech.

Moreover, hate speech (UN-hatedefn) is defined between two parties, *attacker* and *target*. However, inciting speech involves an instrument (non-target) to be angry or take an action against the target. Many studies focus on the automatic identification of hate speech from text (Saha et al. 2021; Aluru et al. 2021; Das et al. 2022; Bohra et al. 2018; Gambäck and Sikdar 2017; Zhang et al. 2018; Nobata et al. 2016; Park and Fung 2017; Del Vigna et al. 2017) but inciting speech's identification is not well-studied.

4.7.2 Dangerous Speech

Dangerous speech refers to the text that can invoke violence against a group[3]. Such violence-invoking cases also overlap with inciting speech involving exhortation. For example, *"Someday Hindus should be ready to fight against Muslims."* falls into both dangerous speech and exhortation. However, the exhortation is not limited to violence but also includes cases of supporting anti-target groups and discriminating against the target's possessions. For example, *"You buy only from Hindus like all your festivals, etc."*, asks readers to buy goods only from Hindus and not other religious groups. In India, since Muslims are the next largest religious group after Hindus (Saha et al. 2021), such sentences indirectly urge readers to go against Muslim businesses but don't fall under the umbrella of dangerous speech.

4.7.3 Fear Speech and Islamophobia

The dictionary meaning of Islamophobia is fear of, dislike and hate toward, and discrimination against Islam and Muslims[4]. However, most of the studies on Islamophobia only consider the hate aspect (Sindoni 2018; Vidgen and Yasseri 2020). Only a few studies focus on the fear

[3] `https://dangerousspeech.org/faq/?faq=200`
[4] `https://www.merriam-webster.com/dictionary/Islamophobia`

aspect (Saha et al. 2021, 2023). Saha et al. (Saha et al. 2021) conduct a large-scale study on fearful messages posted on WhatsApp. They curate 27k posts from Indian public WhatsApp groups and find ∼8,000 of them to be fearful through manual annotation and similarity hashing (Gionis et al. 1999). Further, they train models to identify such fearful messages. We leverage Saha et al.'s (Saha et al. 2021) dataset to find inciting sentences that either induce anger or call for some action. We focus on a different emotion (anger instead of fear) evoked against Muslims. We acknowledge that sometimes the same sentence may induce anger and fear in different individuals. However, many inciting sentences such as "You call Jinnah great . . . Be ashamed of something" predominantly instill anger and don't overlap with fear speech. In addition, fear speech doesn't cover exhortation expressed through text.

To the best of our knowledge, we are the first ones to study the pragmatics of inciting speech, curate a labeled dataset of inciting sentences, and explore natural language models for identifying such speech against Muslims.

4.8 Limitations and Future Work

First, our model is trained on sentences that incite readers against Indian Muslims. In the future, we can expand our model to identify incitement sentences against other religious groups such as Hindus, Sikhs, and so on. Second, we leveraged sentences from only one social media platform, WhatsApp. We can expand INCITE to include sentences from multiple platforms (such as Reddit and Twitter) and train cross-platform-based models for identification. Further, NLP models can be built with a broad vision of identifying all three: inciting speech, fear speech, and hate speech. Such models will serve as a one-stop solution to eliminate all the disturbing and targeted text from social media platforms.

CHAPTER

5

UNDERSTANDING VIOLATION OF CONSENT

In Chapter 1, we discussed nine criteria of consent formulated by Singh (2022). These criteria are formed using Habermas's validity claims (Habermas 1984) and analyze consent in terms of both mental and normative acts. We plan to study if the violation of any of these criteria can be identified through the language of the post.

Example 16 shows a harassment-related post from METHREE dataset that implicitly expresses how consent is violated. First, the survivor is a minor, hence it's a violation of statutes criteria. Second, by saying "I didn't know it was wrong", the survivor admits that they were incapable to decide what was right for them. Hence, it is also a violation of capacity criteria.

We illustrate Example 17 to show how consent violation is implicitly expressed in the online harassment post. The same example expresses the power dynamics between the parent and the kid (victim). The victim here even mentions that they were made to get this app. This indirectly shows that the incident is against the kid's free will. Hence, it qualifies for violation of free will and power criteria of consent.

We propose the following research question.

$RQ_{\text{consent-violation}}$: How can we leverage natural language to identify consent violations expressed in harassment posts?

Example 16: Violation of consent expressed in MeToo post

Is this sexual harassment?

When I was <minor-age>, a <adult-age> year-old man used to visit me twice a week. He was my piano teacher. He used to touch me and my clothes. Although he never touched my private part, but sometimes would touch my butt over my underwear. He used to go under my pants many times. He would feel my ribs a lot. I didn't report for months because I didn't know it was wrong but I felt a little weird. I got him fired by telling my mom. Does this count as sexual harassment?

Example 17: Violation of consent expressed in app reviews

This app basically ruined my family to an extent

(for the Life360 app)

"My mother made everyone in the family get this app. She freaks out when the app doesn't do its job because of random obstacles that mess with the location accuracy. Drains the battery and makes my parents paranoid to know where I am at all times. I don't even do any bad stuff, yet years of trust building are being swept away by the ability to spy on the children of a household. If you're a parent I highly recommend you don't get this app because it is extremely uncomfortable to have and it makes parents trust their children less."

Since consent violation is implicitly expressed in harassment posts, solving this problem is non-trivial. We plan to analyze the textual features to identify the violated consent criteria. A computational model will help in the automatic understanding of how consent is violated in each story.

REFERENCES

Larry Alexander. Incitement and Freedom of Speech. In *Freedom of Speech and Incitement Against Democracy*, pages 101–118. Brill Nijhoff, 2001.

Sai Saketh Aluru, Binny Mathew, Punyajoy Saha, and Animesh Mukherjee. A Deep Dive into Multilingual Hate Speech Classification. In *Proceedings of the Machine Learning and Knowledge Discovery in Databases. Applied Data Science and Demo Track: European Conference, ECML PKDD*, pages 423–439, Ghent, Belgium, 2021. Springer International Publishing.

Nazanin Andalibi, Oliver Haimson, Munmun Choudhury, and Andrea Forte. Understanding social media disclosures of sexual abuse through the lenses of support seeking and anonymity. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 3906–3918, San Jose, California, May 2016. Association for Computing Machinery.

Debasish Basak, Srimanta Pal, and Dipak Patranabis. Support Vector Regression. *Neural Information Processing, Letters and Reviews*, 11, November 2007.

Tobias Bauer, Emre Devrim, Misha Glazunov, William Lopez Jaramillo, Balaganesh Mohan, and Gerasimos Spanakis. #metoomaastricht: Building a chatbot to assist survivors of sexual harassment. In *Machine Learning and Knowledge Discovery in Databases*, pages 503–521, Cham, September 2020. Springer International Publishing.

Rosanna Bellini, Emily Tseng, Nora McDonald, Rachel Greenstadt, Damon McCoy, Thomas Ristenpart, and Nicola Dell. So-called privacy breeds evil: Narrative justifications for intimate partner surveillance in online forums. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW3), January 2021. doi: 10.1145/3432909.

Andrew R. Besmer, Jason Watson, and M. Shane Banks. Investigating User Perceptions of Mobile App Privacy: An Analysis of User-Submitted App Reviews. *International Journal of Information Security and Privacy (IJISP)*, 14(4):74–91, October 2020.

Cheng Bo, Guobin Shen, Jie Liu, Xiang-Yang Li, YongGuang Zhang, and Feng Zhao. Privacy.Tag: Privacy Concern Expressed and Respected. In *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems*, pages 163–176, New York, November 2014. Association for Computing Machinery.

Aditya Bohra, Deepanshu Vijay, Vinay Singh, Syed Sarfaraz Akhtar, and Manish Shrivastava. A Dataset of Hindi-English Code-Mixed Social Media Text for Hate Speech Detection. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, pages 36–41, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

Hanen Borchani, Gherardo Varando, Concha Bielza, and Pedro Larrañaga. A Survey on Multi-Output Regression. *Wiley Interdisciplinary Reviews Data Mining and Knowledge Discovery*, 5 (5):216–233, September 2015.

Sandrine Boudana and Elad Segev. Theorizing Provocation Narratives as Communication Strategies: Provocation Narratives. *Communication Theory*, 27:329–346, 06 2017.

Denis Eka Cahyani and Irene Patasik. Performance Comparison of TF-IDF and Word2Vec Models for Emotion Text Classification. *Bulletin of Electrical Engineering and Informatics*, 10(5):2780–2788, 2021.

Clay Calvert. Hate Speech and Its Harms: A Communication Theory Perspective. *Journal of Communication*, 47(1):4–19, 1997.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. Universal Sentence Encoder. *CoRR*, abs/1803.11175:1–7, 2018.

Jair Cervantes, Farid Garcia-Lamont, Lisbeth Rodríguez-Mazahua, and Asdrubal Lopez. A Comprehensive Survey on Support Vector Machine Classification: Applications, Challenges and Trends. *Neurocomputing*, 408:189–215, 2020.

Rahul Chatterjee, Periwinkle Doerfler, Hadas Orgad, Sam Havron, Jackeline Palmer, Diana Freed, Karen Levy, Nicola Dell, Damon McCoy, and Thomas Ristenpart. The Spyware Used in Intimate Partner Violence. In *Proceedings of the 39th IEEE Symposium on Security and Privacy (SP)*, pages 441–458, San Francisco, CA, USA, May 2018. IEEE Press.

Jacob Cohen. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960.

Jacob Cohen. *Statistical Power Analysis for the Behavioral Sciences*. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 2nd edition, 1988.

Aron Culotta and Andrew McCallum. Reducing Labeling Effort for Structured Prediction Tasks. In *AAAI*, volume 5, pages 746–751, 2005.

Jonathan Culpeper. *Impoliteness: Using Language to Cause Offence*, volume 28. Cambridge University Press, 2011.

Jonathan Culpeper, Derek Bousfield, and Anne Wichmann. Impoliteness Revisited: With Special Reference to Dynamic and Prosodic Aspects. *Journal of Pragmatics*, 35(10-11):1545–1579, 2003.

Ido Dagan and Sean P Engelson. Committee-based Sampling for Training Probabilistic Classifiers. In *Machine Learning Proceedings*, pages 150–157. Elsevier, 1995.

Mithun Das, Somnath Banerjee, and Animesh Mukherjee. Data Bootstrapping Approaches to Improve Low Resource Abusive Language Detection for Indic Languages. In *Proceedings of the 33rd ACM Conference on Hypertext and Social Media*, pages 32–42, New York, NY, USA, 2022. Association for Computing Machinery.

Bonnie-Elene Deal, Lourdes S Martinez, Brian H Spitzberg, and Ming-Hsiang Tsou. "I Definitely Did Not Report It When I Was Raped...# WeBelieveChristine# MeToos": A Content Analysis of Disclosures of Sexual Assault on Twitter. *Social Media+ Society*, 6(4):2056305120974610, 2020.

Fabio Del Vigna, Andrea Cimino, Felice Dell'Orletta, Marinella Petrocchi, and Maurizio Tesconi. Hate Me, Hate Me Not: Hate Speech Detection on Facebook. In *Proceedings of the First Italian Conference on Cybersecurity (ITASEC17)*, pages 86–95, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

Stephan Dreiseitl and Lucila Ohno-Machado. Logistic Regression and Artificial Neural Network Classification Models: a Methodology Review. *Journal of Biomedical Informatics*, 35(5): 352–359, 2002.

Mai ElSherief, Vivek Kulkarni, Dana Nguyen, William Yang Wang, and Elizabeth Belding. Hate lingo: A Target-based Linguistic Analysis of Hate Speech in Social Media. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 12, 2018.

Matthias Fassl, Simon Anell, Sabine Houy, Martina Lindorfer, and Katharina Krombholz. Comparing User Perceptions of Anti-Stalkerware Apps with the Technical Reality. In *Proceedings of the Symposium on Usable Privacy and Security (SOUPS)*, pages 135–154, Boston, MA, August 2022. USENIX Association.

Anjalie Field, Gayatri Bhat, and Yulia Tsvetkov. Contextual Affective Analysis: A Case Study of People Portrayals in Online #Metoo Stories. In *Proceedings of the 13th International AAAI Conference on Web and Social Media*, volume 13, pages 158–169, 2019.

Kimberly Field-Springer, Hannah Draut, Fran Babrow, and Maddie Sandman. (Re)claiming Stories in the #MeToo Movement: Righting Epistemic Wrongs of Physical, Mental, and Emotional Harms of Sexual Violence. *Health Communication*, 0(0):1–10, February 2021. doi: 10.1080/10410236.2021.1880052.

Diana Freed, Jackeline Palmer, Diana Minchala, Karen Levy, Thomas Ristenpart, and Nicola Dell. "A Stalker's Paradise": How Intimate Partner Abusers Exploit Technology. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–13, New York, April 2018. Association for Computing Machinery.

Diana Freed, Sam Havron, Emily Tseng, Andrea Gallardo, Rahul Chatterjee, Thomas Ristenpart, and Nicola Dell. Is My Phone Hacked? Analyzing Clinical Computer Security Interventions with Survivors of Intimate Partner Violence. *Proceedings of the 17th ACM Conference on Human-Computer Interaction*, 3:1–24, 2019.

Björn Gambäck and Utpal Kumar Sikdar. Using Convolutional Neural Networks to Classify Hate-Speech. In *Proceedings of the First Workshop on Abusive Language Online*, pages 85–90, Vancouver, BC, Canada, August 2017. Association for Computational Linguistics.

Akash Gautam, Puneet Mathur, Rakesh Gosangi, Debanjan Mahata, Ramit Sawhney, and Rajiv Ratn Shah. #MetooMA: Multi-Aspect Annotations of Tweets Related to the Metoo Movement. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 14, pages 209–216, 2020.

Theodore Georgiou, Amr El Abbadi, and Xipeng Yan. Privacy Cyborg: Towards Protecting the Privacy of Social Media Users. In *Proceedings of 33rd International Conference on Data Engineering (ICDE)*, pages 1395–1396, San Diego, April 2017. IEEE Computer Society.

Arijit Ghosh Chowdhury, Ramit Sawhney, Puneet Mathur, Debanjan Mahata, and Rajiv Ratn Shah. Speak up, Fight Back! Detection of Social Media Disclosures of Sexual Harassment. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, Minneapolis, Minnesota, June 2019a. Association for Computational Linguistics.

Arijit Ghosh Chowdhury, Ramit Sawhney, Rajiv Ratn Shah, and Debanjan Mahata. #YouToo? Detection of Personal Recollections of Sexual Harassment on Social Media. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2527–2537, Florence, Italy, July 2019b. Association for Computational Linguistics.

Aristides Gionis, Piotr Indyk, Rajeev Motwani, et al. Similarity Search in High Dimensions via Hashing. In *Proceedings of the Very Large Data Bases*, volume 99, pages 518–529, 1999.

Andrej Gisbrecht, Alexander Schulz, and Barbara Hammer. Parametric Nonlinear Dimensionality Reduction Using Kernel t-SNE. *Neurocomputing*, 147:71–82, 2015.

Alessandra Gorla, Ilaria Tavecchia, Florian Gross, and Andreas Zeller. Checking App Behavior Against App Descriptions. In *Proceedings of the 36th International Conference on Software Engineering*, pages 1025–1035, New York, NY, USA, May 2014. Association for Computing Machinery.

Jürgen Habermas. *The Theory of Communicative Action, Volumes 1 and 2*. Polity press, Cambridge, United Kingdom, 1984.

Kevin A. Hallgren. Computing Inter-Rater Reliability for Observational Data: An Overview and Tutorial. *Tutorials in Quantitative Methods for Psychology*, 8(1):23–34, 2012.

Steve Hanneke. Theory of Active Learning. *Foundations and Trends in Machine Learning*, 7 (2–3), 2014.

Hamza Harkous, Kassem Fawaz, Rémi Lebret, Florian Schaub, Kang G. Shin, and Karl Aberer. Polisis: Automated Analysis and Presentation of Privacy Policies Using Deep Learning. In *Proceedings of 27th USENIX Security Symposium*, pages 531–548, Baltimore, August 2018. USENIX Association.

Rakibul Hasan, David Crandall, Mario Fritz, and Apu Kapadia. Automatically Detecting Bystanders in Photos to Reduce Privacy Risks. In *IEEE Symposium on Security and Privacy (SP)*, Los Alamitos, May 2020. IEEE Computer Society.

Naeemul Hassan, Amrit Poudel, Jason Hale, Claire Hubacek, Khandaker Tasnim Huq, Shubhra Kanti Karmaker Santu, and Syed Ishtiaque Ahmed. Towards Automated Sexual Violence Report Tracking. In *Proceedings of the 14th International AAAI Conference on Web and Social Media*, pages 250–259, Georgia, USA, June 2020.

Sam Havron, Diana Freed, Rahul Chatterjee, Damon McCoy, Nicola Dell, and Thomas Ristenpart. Clinical Computer Security for Victims of Intimate Partner Violence. In *Proceedings of the 28th USENIX Security Symposium*, pages 105–122, Santa Clara, January 2019. USENIX Association.

Benjamin Henne, Christian Szongott, and Matthew Smith. SnapMe If You Can: Privacy Threats of Other Peoples' Geo-Tagged Media and What We Can Do About It. In *Proceedings of the 6th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 95–106, New York, May 2013. Association for Computing Machinery.

Wesley Newcomb Hohfeld. *Fundamental Legal Conceptions as Applied in Judicial Reasoning: and other Legal Essays*. Yale University Press, 1923.

Sweta Karlekar and Mohit Bansal. SafeCity: Understanding Diverse Forms of Sexual Harassment Personal Stories. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2805–2811, Brussels, Belgium, October 2018. Association for Computational Linguistics.

Aparup Khatua, Erik Cambria, and Apalak Khatua. Sounds of Silence Breakers: Exploring Sexual Violence on Twitter. In *Proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 397–400, Barcelona, 2018. IEEE Press.

Felix Koch. Consent as a Normative Power. In *The Routledge Handbook of the Ethics of Consent*, pages 32–43. Routledge, 2018.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128:1–26, 2020. doi: 10.1007/s11263-020-01316-z.

Yingchi Liu, Quanzhi Li, Xiaozhong Liu, Qiong Zhang, and Luo Si. Sexual harassment story classification and key information identification. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2385–2388, New York, NY, USA, November 2019a. Association for Computing Machinery.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A Robustly Optimized Bert Pretraining Approach. *arXiv preprint arXiv:1907.11692*, 2019b.

Piotr Łubiński. Social Media Incitement to Genocide: ECHR Countries Perspective. In *The Concept of Genocide in International Criminal Law*. Taylor & Francis, 2017.

Shah Mahmood and Yvo Desmedt. Your facebook deactivated friend or a cloaked spy. In *Proceedings of the 33rd IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pages 367–373, Los Alamitos, March 2012. IEEE Computer Society.

Lydia Manikonda, Ghazaleh Beigi, Subbarao Kambhampati, and Huan Liu. #metoo through the lens of social media. In *Proceedings of the 11th International Conference on Social, Cultural, and Behavioral Modeling*, pages 104–110, Washington, USA, November 2018. Springer Verlag.

Christopher D. Manning. Part-of-speech tagging from 97% to 100%: Is it time for some linguistics? In *Proceedings of the Computational Linguistics and Intelligent Text Processing*, pages 171–189, Berlin, Heidelberg, February 2011. Springer.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. Hatexplain: A Benchmark Dataset for Explainable Hate Speech Detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14867–14875, 2021.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS, pages 3111–3119, Lake Tahoe, Nevada, December 2013. Neural Information Processing Systems Foundation.

George A. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 38(11): 39–41, November 1995. doi: 10.1145/219717.219748.

Saif Mohammad and Peter Turney. Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 26–34, Los Angeles, California, June 2010. Association for Computational Linguistics.

Saif M. Mohammad and Peter D. Turney. Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3):436–465, 2013.

Duc Cuong Nguyen, Erik Derr, Michael Backes, and Sven Bugiel. Short text, large effect: Measuring the impact of user reviews on android app security & privacy. In *Proceedings of the 40th IEEE Symposium on Security and Privacy (SP)*, pages 555–569, San Francisco, May 2019. IEEE Computer Society.

Nltk. NLTK Library. `https://www.nltk.org/api/nltk.tokenize.html`.

Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. Abusive Language Detection in Online User Content. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, pages 145–153, Republic and Canton of Geneva, CHE, 2016. International World Wide Web Conferences Steering Committee.

NortonApp. Nortan mobile security app. https://buy-static.norton.com/norton/ps/bb/ushard/4up_mnav05w_us_en_fl_tw_branded_mix-n360.html.

Lorelli S. Nowell, Jill M. Norris, Deborah E. White, and Nancy J. Moules. Thematic Analysis: Striving to Meet the Trustworthiness Criteria. *International Journal of Qualitative Methods*, 16(1):1609406917733847, 2017.

Tully O'Neill. 'today i speak': Exploring how victim-survivors use reddit. *International Journal for Crime, Justice and Social Democracy*, 7(1):44, 2018.

Onevsone. One versus One Approach. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html.

Rahul Pandita, Xusheng Xiao, Wei Yang, William Enck, and Tao Xie. WHYPER: Towards automating risk assessment of mobile applications. In *Proceedings of the 22nd USENIX Security Symposium*, pages 527–542, Washington, D.C., USA, August 2013. USENIX Association.

Ji Ho Park and Pascale Fung. One-step and Two-step Classification for Abusive Language Detection on Twitter. In *Proceedings of the First Workshop on Abusive Language Online*, pages 41–45, Vancouver, BC, Canada, August 2017. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global Vectors for Word Representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics.

Juan Carlos Pereira-Kohatsu, Lara Quijano-Sánchez, Federico Liberatore, and Miguel Camacho-Collados. Detecting and Monitoring Hate Speech in Twitter. *Sensors*, 19(21):4654, 2019.

Alfredo J. Perez, Sherali Zeadally, and Scott Griffith. Bystanders' privacy. *IT Professional*, 19(3): 61–65, 2017. doi: 10.1109/MITP.2017.42.

Zhengyang Qu, Vaibhav Rastogi, Xinyi Zhang, Yan Chen, Tiantian Zhu, and Zhong Chen. Autocog: Measuring the description-to-permission fidelity in android applications. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pages 1354–1365, New York, NY, USA, November 2014. Association for Computing Machinery.

Jake Reichel, Fleming Peck, Mikako Inaba, Bisrat Moges, Brahmnoor Singh Chawla, and Marshini Chetty. 'i have too much respect for my elders': Understanding south african mobile users' perceptions of privacy and current behaviors on facebook and whatsapp. In *Proceedings of the 29th USENIX Security Symposium*, pages 1949–1966, Virtual, August 2020. USENIX Association.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics.

Ana Reyes-Menendez, Jose Ramon Saura, and Stephen B. Thomas. Exploring key indicators of social identity in the #metoo era: Using discourse analysis in ugc. *International Journal of Information Management*, 54:102129, 2020.

Kevin A. Roundy, Paula Barmaimon Mendelberg, Nicola Dell, Damon McCoy, Daniel Nissani, Thomas Ristenpart, and Acar Tamersoy. The many kinds of creepware used for interpersonal attacks. In *Proceedings of the 41th IEEE Symposium on Security and Privacy (SP)*, pages 753–770, Los Alamitos, May 2020. IEEE Computer Society.

Punyajoy Saha, Binny Mathew, Kiran Garimella, and Animesh Mukherjee. "Short is the Road That Leads from Fear to Hate": Fear Speech in Indian Whatsapp Groups. In *Proceedings of the Web Conference*, pages 1110–1121, New York, 2021. Association for Computing Machinery.

Punyajoy Saha, Kiran Garimella, Narla Komal Kalyan, Saurabh Kumar Pandey, Pauras Mangesh Meher, Binny Mathew, and Animesh Mukherjee. On the Rise of Fear Speech in Online Social Media. *Proceedings of the National Academy of Sciences*, 120(11):e2212270120, 2023.

Joni Salminen, Maximilian Hopf, Shammur A Chowdhury, Soongyo Jung, Hind Almerekhi, and Bernard J Jansen. Developing an Online Hate Classifier for Multiple Social Media Platforms. *Human-centric Computing and Information Sciences*, 10:1–34, 2020.

Claude Sammut and Geoffrey I. Webb, editors. *Mean Squared Error*, pages 653–653. Springer US, Boston, MA, 2010. ISBN 978-0-387-30164-8. doi: 10.1007/978-0-387-30164-8_528.

Kimberly T Schneider and Nathan J Carpenter. Sharing# metoo on twitter: Incidents, coping responses, and social reactions. *Equality, Diversity and Inclusion: An International Journal*, 2019.

Andrew Sellars. Defining Hate Speech. *Berkman Klein Center Research Publication*, (2016-20): 16–48, 2016.

Burr Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114, 2012.

Maria Grazia Sindoni. Direct Hate Speech Vs. Indirect Fear Speech. A Multimodal Critical Discourse Analysis of the Sun's Editorial "1 in 5 Brit Muslims' Sympathy for Jihadis". *Lingue e Linguaggi*, 28:267–292, 2018.

Munindar P Singh. Consent as a Foundation for Responsible Autonomy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 12301–12306, 2022.

Paul F. Smith, Siva Ganesh, and Ping Liu. A comparison of random forest regression and multiple linear regression for prediction in neuroscience. *Journal of Neuroscience Methods*, 220:85–91, October 2013.

Susan Benesch. Defining and Diminishing Hate Speech. *State of the World's Minorities and Indigenous Peoples*, 2014:18–25, 2014.

Emily Tseng, Rosanna Bellini, Nora McDonald, Matan Danos, Rachel Greenstadt, Damon McCoy, Nicola Dell, and Thomas Ristenpart. The tools and tactics used in intimate partner surveillance: An analysis of online infidelity forums. In *Proceedings of the 29th USENIX Security Symposium*, pages 1893–1909, Virtual, August 2020. USENIX Association.

Emily Tseng, Diana Freed, Kristen Engel, Thomas Ristenpart, and Nicola Dell. A digital safety dilemma: Analysis of computer-mediated computer security interventions for intimate partner violence during covid-19. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–17, Virtual, 2021. Association for Computing Machinery.

Emily Tseng, Mehrnaz Sabet, Rosanna Bellini, Harkiran Kaur Sodhi, Thomas Ristenpart, and Nicola Dell. Care infrastructures for digital security in intimate partner violence. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, New Orleans, 2022. Association for Computing Machinery. doi: 10.1145/3491102.3502038.

UN-hatedefn. United Nations on Hate Speech. `https://www.un.org/en/hate-speech/understanding-hate-speech/what-is-hate-speech`.

Alper Kursat Uysal and Serkan Gunal. The impact of preprocessing on text classification. *Information Processing & Management*, 50(1):104–112, 2014.

Bertie Vidgen and Taha Yasseri. Detecting Weak and Strong Islamophobic Hate Speech on Social Media. *Journal of Information Technology & Politics*, 17(1):66–78, 2020.

Min Xu, Pakorn Watanachaturaporn, Pramod K. Varshney, and Manoj K. Arora. Decision tree regression for soft classification of remote sensing data. *Remote Sensing of Environment*, 97 (3):322–336, 2005. doi: https://doi.org/10.1016/j.rse.2005.05.008.

Peng Yan, Linjing Li, Weiyun Chen, and Daniel Zeng. Quantum-inspired density matrix encoder for sexual harassment personal stories classification. In *2019 IEEE International Conference on Intelligence and Security Informatics, ISI*, pages 218–220, Shenzhen, China, July 2019. Institute of Electrical and Electronics Engineers Inc.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. XLNet: Generalized Autoregressive Pretraining for Language Understanding. *Advances in Neural Information Processing Systems*, 32, 2019.

William J Youden. Index for rating diagnostic tests. *Cancer*, 3(1):32–35, 1950.

Mohammed Zakariah. Classification of Large Datasets Using Random Forest Algorithm in Various Applications: Survey. *International Journal of Engineering and Innovative Technology (IJEIT)*, 4:189–198, September 2014.

Ziqi Zhang, David Robinson, and Jonathan Tepper. Detecting Hate Speech on Twitter Using a Convolution-Gru Based Deep Neural Network. In *European Semantic Web Conference*, pages 745–760. Springer, 2018.

Sebastian Zimmeck, Ziqi Wang, Lieyong Zou, Roger Iyengar, Bin Liu, Florian Schaub, Shomir Wilson, Norman Sadeh, Steven Bellovin, and Joel Reidenberg. Automated analysis of privacy requirements for mobile apps. In *Proceedings of the Network and Distributed System Security Symposium*, San Diego, February 2017. Korea Society of Internet Information.

Yixin Zou, Allison McDonald, Julia Narakornpichit, Nicola Dell, Thomas Ristenpart, Kevin Roundy, Florian Schaub, and Acar Tamersoy. The role of computer security customer support in helping survivors of intimate partner violence. In *Proceedings of the 30th USENIX Security Symposium*, pages 429–446, Virtual, 2021. USENIX Association.

APPENDICES