Privacy Protection in Conversations

Qiongkai Xu

A thesis submitted for the degree of
Doctor of Philosophy of
The Australian National University

August 2022

© Qiongkai Xu 2022

Except where otherwise indicated, this thesis is my own original work.

Qiongkai Xu
25 August 2022

to my family and friends

Acknowledgments

I would like to express my gratitude towards everyone who made this thesis possible.

First and foremost, an enormous thank you to my primary supervisor, Lizhen Qu, for his guidance, support, and encouragement throughout my research for this thesis. I have learned a lot from him about how to conduct research. I am also grateful to Lizhen for his kind support during my hard times.

Thank you to Lexing Xie and Richard Nock for their guidance and advice in my early work. Thank you to Gholamreza Haffari, Yolande Strengers, Chenchen Xu, Jarrod Knibbe, and other collaborators for their insights and contributions that help to form this thesis.

I am thankful to the Australian National University and Data61 CSIRO (previously NICTA) for providing financial and technical support for my research. I am also thankful to Monash University for their technical support on part of my research.

Thank you to all the members of the ANU Computational Media Lab and the Monash Vision and Language Group for providing excellent environments for research and study. Thank you to the friends and colleagues in Australia who made my time during the thesis a memorable one.

Finally, thank you to my family. None of these would have been possible without your support. Thank you to my wife for understanding and supporting of my research dream.

Abstract

Leakage of personal information in online conversations raises serious privacy concerns. For example, malicious users might collect sensitive personal information from vulnerable users via deliberately designed conversations. This thesis tackles the problem of privacy leakage in textual conversations and proposes to mitigate the risks of privacy disclosure by *detecting* and *rewriting* the risky utterances. Previous research on privacy protection in text has a focus on manipulating the implicit semantic representations in a continuous high dimensional space, which are mostly used for eliminating trails of personal information to machine learning models. Our research has a focus on the explicit expressions of conversations, namely sequences of words or tokens, which are generally used between human interlocutors or human-computer interactions. The new setting for privacy protection in text could be applied to the conversations by individual human users, such as vulnerable people, and artificial conversational bots, such as digital personal assistants.

This thesis consists of two parts, essentially answering two research questions: *How to detect the utterances with the risk of privacy leakage?* and *How to modify or rewrite the utterances into the ones with less private information?*

In the first part of this thesis, we aim to detect the utterances with privacy leakage risk and report the sensitive utterances to authorized users for approval. One of the essential challenge of the detection task is that we cannot acquire a large-scale aligned corpus for supervised training of natural language inference for private information. A compact dataset is collect to merely validate the privacy leakage detection models. We investigate weakly supervised methods to learn utterance-level inference from coarse set-level alignment signals. Then, we propose novel alignment models, *i.e.*, Sharp-Max and Sparse-Max, for utterance inference. Our approaches manage to outperform competitive baseline alignment methods. Additionally, we develop a privacy-leakage detection system integrated in Facebook Messenger to demonstrate the utility of our proposed task in real-world usage scenarios.

In the second part of this thesis, we investigate two pieces of work to rewrite the privacy-leakage sentences automatically into less sensitive ones. The first work discusses obscuring personal information in form of classifiable attributes. We propose to reduce the bias of sensitive attributes, such as gender, political slant and race, using an obscured text rewriting models. The rewriting models are guided by corresponding classifiers for the personal attributes to obscure. Adversarial training and fairness risk measurement are proposed to enhance the fairness of the generators, alleviating privacy leakage of the target attributes. The second work protects personal information in the form of open-domain textual descriptions. We further explore three feasible rewriting strategies, *deleting*, *obscuring*, and *steering*, for privacy-aware

text rewriting. We investigate the possibility of fine-tuning a pre-trained language model for privacy-aware text rewriting. Based on our dataset, we further observe the relation of rewriting strategies to their semantic spaces in a knowledge graph. Then, a simple but effective decoding method is developed to incorporate these semantic spaces into corresponding rewriting models.

As a whole, this thesis presents a comprehensive study and the first solutions in varying settings for protecting privacy in conversations. We demonstrate that both privacy leakage detection and privacy-aware text rewriting are plausible using machine learning methodologies. Our contributions also include novel ideas for text alignment for natural language inference, training technologies for text attribute obfuscating, and open-domain knowledge guidance to text rewriting. This thesis opens up inquiries into protecting sensitive user information in conversations from the perspective of explicit text representation.

Publications, Software & Data

The following original publications were made during the development of this thesis as part of the Doctor of Philosophy programme. Software and data produced for these publications is provided to form a basis for future work.

Publications

Xu, Q., Qu, L., Xu, C., & Cui, R. (2019). Privacy-aware text rewriting. In Proceedings of the 12th International Conference on Natural Language Generation, 247-257.
https://www.inlg2019.com/assets/papers/76_Paper.pdf

Xu, Q., Xu, C., & Qu, L. (2019). ALTER: Auxiliary Text Rewriting Tool for Natural Language Generation. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing: System Demonstrations, 13-18.
https://www.aclweb.org/anthology/D19-3003.pdf

Strengers, Y., Qu, L., **Xu, Q.**, & Knibbe, J. (2020). Adhering, Steering, and Queering: Treatment of Gender in Natural Language Generation. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, 1-14.
https://doi.org/10.1145/3313831.3376315

Xu, Q., Qu, L., Gao, Z., & Haffari, G. (2020). Personal Information Leakage Detection in Conversations. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, 6567-6580.
https://www.aclweb.org/anthology/2020.emnlp-main.532.pdf

Xu, Q., Xu, C., & Qu, L. (2021). Privacy Monitoring Service for Conversations. In Proceedings of the 14th ACM International Conference on Web Search and Data Mining: System Demonstrations, 1093-1096.
https://doi.org/10.1145/3437963.3441706

Xu, Q., Qu, L., Hou, X., & Haffari, G. (2021). Open-Domain Privacy-Aware Text Rewriting Using Specific Strategies.

Software, Data, & Demo

PILD (Personal Information Leakage Detection): source code and dataset.
https://github.com/xuqiongkai/PILD

PMS (Privacy Monitor System): demo.
https://drive.google.com/file/d/131Zq94FH1gi1EITqesnz9cGVsbdR7ttF/view

PATR (Privacy-Aware Text Rewriting): datasets.
https://github.com/xuqiongkai/PATR

ALTER (Auxiliary Text Rewriting Tool): source code, models and demo.
https://github.com/xuqiongkai/ALTER
https://drive.google.com/file/d/1GS-kbhhJKMxM3pxBZJdn7sVxP9rAfe4w/view

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Personal information could be revealed in conversations, from human users, their relatives, their friends or their owned devices. Unconscious information leakage could result in incalculable consequences, *e.g.*, the phishing attacks by malicious interlocutors who utilize the disclosed information to forge their relationship to the victims. In this thesis, we propose to protect the privacy of users by mitigate the disclosure of their sensitive personal information.

In the modern world, more and more conversations are conducted via digital platforms or devices and privacy disclosure may occur in different type of conversations, such as human-human and human-bot conversations. The most common type of conversation is *human-human* conversation between two human users. Human might accidentally and inattentively disclose their personal information to other (malicious) user [Lipford et al., 2008; Schofield and Joinson, 2008; Chawdhry et al., 2013]. In addition, *privacy paradox* phenomenon states that even users with high level of privacy concerns do not always take appropriate actions although those measures are fairly easy to perform [Norberg et al., 2007]. People in vulnerable groups are especially fragile to privacy leakage issues. For example, children are more likely to unconsciously disclose their sensitive information. Another type is *human-bot* conversation conducted between human and bots. Due to the rapid growth of smart speakers or home robots,[1] the human-bot conversations poses more and more ratios of conversations in our daily lives. The digital devices or bots of the users also pose serious privacy concerns, as the recent advanced dialogue models are mostly trained as black-box models and we have limited understanding and control of their behaviors. These models could disclose their owners' sensitive information to the out-side interlocutors, including malicious attackers. As a result, there is an urgent requirement for protecting personal information in conversations for both (vulnerable) people and their bot agents.

1.1 Privacy Protection on Explicit Text

Privacy preservation for data has a long history spanning multiple disciplines. One of the popular research directions is privacy preservation in algorithm which intends

[1] https://www.opus.global/media/44137/opus-q3-2018-report-eng.pdf

to eliminate the influence of sensitive information in an algorithm. For example, the decisions from a machine learning classifiers do not discriminate users from different groups according to their gender or age. Another example is encoding features, including sensitive personal attributes, into high-dimensional representations that cannot be confidently categorized into sensitive groups by adversarial classifiers. Most of these algorithms are motivated and utilized by institutions, such as governments, research organizations, commercial companies and etc., for various purposes, such as decision-making, advertisement recommendation, and providing personalized services. In these cases, the private information of individual users is assumed to be collected, stored, processed and protected properly by those institutions. Changing the perspective from institutions to individuals, we unfortunately cannot guarantee that all institutions and companies will endeavor to provide proper privacy protection components for each individual user. In even worse cases, *i)* "*Data Leakage*" incidents could be caused or perpetrated by insiders [Taal et al., 2017] and *ii)* data privacy issues are normally associated with company merger and acquisition, especially cross-border transfers of information or other activity with cross-border implications [Sherer et al., 2015]. Hence, your private data could be leaked by these institutions or misused by untrustworthy third parties.We suggest that privacy disclosure in text, such as online conversations and personal tweets, could be prevented before they are sent out and collected by these institutions. The new privacy protection systems should work on behalf of individuals and ideally run independently on individual users' own devices. We consider both human users and robotic assistants as individuals, who have requirements for privacy protection. The privacy protection for individuals should focus on the explicit representation of the data, or rather *texts* in this thesis, as the sensitive information is more naturally encoded in the original textual expressions. We further demonstrate some more detailed usage scenarios of privacy protection for individual users with regard to both human and bot.

(a) **Preventing Scams**: Scams and frauds are targeting everyone.[2] Scammers are getting smarter and taking advantages of new technologies to acquire information to fraud victims [Titus and Gover, 2001]. For example, the criminal can find details of their victim by reviewing social media of a victim and call the back to change the victim's password.[3] The phishing attack also skyrocketed after COVID-19 pandemic and led to an increase in loses [Bitaab et al., 2020; Competition et al., 2020]. Reducing users' personal information disclosure to public could be a way to decrease the risks from scam attacks.

(b) **Digital Personal Assistant**: Recent research on dialogue generation demonstrates the success in improving the quality of utterances by integrating personal information [Zhang et al., 2018b; Liu et al., 2020]. However, giving digital personal assistants the access to their owners' personal information introduces

[2]https://www.scamwatch.gov.au/get-help/protect-yourself-from-scams

[3]https://www.9news.com.au/national/cyber-hackers-criminals-getting-smarter-accessing-bank-accounts/a32121f0-5ba0-492f-b4de-3d27ddb81d7f

the risk that the assistants may disclose the information to malicious interlocutors, when the bots are sent out for tasks. For example, when a personal assistant is asked to buy a meal, the address of the owner could be provided to the cashier involved in this trade, while not to a stranger. Detecting and rewriting the risky utterances helps the bots respond properly.

(c) **Pre-trained Language Model**: The development and deployment of large-scale pre-trained language models have extended the state of the art on a wide array of NLP tasks [Wang et al., 2019]. Despite the great success of these models, the ethical boundaries of these technologies are ambiguous. For example, *i)* pre-trained language model may disclose personal information in the training corpus; *ii)* pre-trained language model may generate unexpected sentences that include toxicity or bias [Pan et al., 2020; Bender et al., 2021; Shen et al., 2021]. These concerns limit the usage of pre-trained language models in real-world applications. Our research could serve as an auxiliary component *i)* to identify the information leakage in the model outputs or the training corpus; and *ii)* to rewrite the generated sentences with unexpected information or patterns to more proper ones.

1.2 Proposed Tasks

In order to protect the privacy leakage from the side of individual users, we focus on the representations of explicit text. We summarize two primary requirements for privacy protection in text:

(a) *Identifying the utterances with privacy leakage risks?* Detecting those risky utterance could potentially be used to alarm users from scams when they intend to disclose their private information to malicious interlocutors, *e.g.*, a juvenile intends to tweet his or her home address and recent family travel arrangements. The digital personal assistants can also be interrupted when they intend to generate inappropriate responses to untrustworthy interlocutors, *e.g.*, providing sensitive personal information of the owners. We use an utterance or a label to represent the sensitive personal information. The information leakage is decided by a text similarity module or a discriminator, respectively.

(b) *Modifying the risky utterances and mitigating the privacy leakage risks.* This function could potentially be used to enhance the current dialogue systems and language models by reducing their privacy concerns, *e.g.*, eliminating the risky information when the dialogue systems generate the owners' names, ages, identity numbers, and *etc.* in their responses. The suggested modifications can also be served as advice to human users when privacy leakage issues are raised, *e.g.*, obscuring the home address and the travel schedule from the tweets.

In alignment with these requirements, we propose to *detect* and *rewrite* the utterances, or sentences, with potential privacy risk. The two steps are correspondent to two modules:

(a) **Detection Module** detects the utterances that disclose personal information, and reports the sensitive ones to the authorized users.

(b) **Rewriting Module** rewrites the sensitive utterances to mitigate the influence of the privacy leakage, given corresponding personal information as control signals.

1.3 Challenges and Key Contributions

In this section, we decompose our tasks into more detailed challenges and list corresponding contributions in this thesis:

- **Problem definition of privacy leakage protection in conversations.** We propose and define a new task of protecting user privacy in their explicit textural expressions, with a focus on conversational usage scenarios. We discuss the emergent requirements of this task and potential usage scenarios. We propose to alleviate the privacy issues by detecting and rewriting the risky texts. (In Chapter 1)
- **How to identify the privacy leakage utterances in conversations?** We first propose to transfer the question as a natural language inference problem, namely identify the utterances that can infer the interlocutor's personal information. Then, we collect the first corpus for evaluation. A weakly supervised alignment framework is proposed to learn the inference model, given coarse alignment signals between utterances and personal information descriptions. Based on this framework, we derive two novel alignment approaches that outperform existing baseline methods. (In Chapter 3)
- **How plausible is the detection approach in real-world applications?** We develop a privacy monitor system that intervenes sensitive messages on a real-world social media platform, *i.e.*, Facebook Messenger. The system demonstrates a way to protect privacy from disclosure using our privacy-leakage detection model. (In Chapter 3)
- **How to obscure the sensitive information utterances?** In our preliminary work on privacy-aware text rewriting, we propose to obscure the classifiable personal attributes in text via rewriting. Given non-parallel corpora with sensitive personal attributes of the writers, we develop privacy-aware text rewriting models based on a back-translation framework. In particular, the attributes are eliminated in back-translation step, through a '*fair*' or '*neutralized*' translator, optimized by adversarial training or fairness-risk measurement as additional constraints with regard to the given sensitive attributes. (In Chapter 4)
- **How to eliminate the influence of the privacy disclosure?** Inspired by the treatment for gender bias in [Strengers et al., 2020],[4] we propose three rewrit-

[4]We consider gender category a special case of sensitive personal information.

ing patterns *deleting*, *obscuring* and *steering* to eliminate the privacy leakage in utterances. We also consider protecting sensitive personal information in the form of open-domain textual descriptions, which is more diverse than classifiable attributes. We collect a compact parallel dataset for this rewriting task. Then, we investigated *i)* fine-tuning pre-trained language model for this tasks; and *ii)* incorporating prior knowledge from semantic knowledge graph into the decoder of our generation model. Our work demonstrates the plausibility of training generation models to modify personal information in text using different patterns. (In Chapter 5)

1.4 Summary

In this thesis, we propose a new research direction of protecting privacy in explicit conversational text. In order to achieve such goal, we propose to detect and rewrite the utterances with privacy risk. The background and related work of this thesis will be introduced in Chapter 2. A systematic solution of privacy leakage detection and its application will be described in Chapter 3. Privacy-aware text rewriting researches will be discussed in Chapter 4 and Chapter 5, based on different settings. Finally, a conclusion of this thesis will be summarized in Chapter 6.

Chapter 2

Background and Related Work

Privacy is an essential human right that consolidates freedom of association, thought and expression, as well as freedom from discrimination [Schofield and Joinson, 2008; aus]. However, information technology not only introduces advantages and convenience to our daily lives, but also increases the risk of disclosing or misusing our personal information. Such right includes being able to control who can view or use information about you. Recently, there has been a lot of work on various directions towards protecting personal information. For example, encryption technology prevents the unauthorized parties or users form accessing sensitive information by converting the original data into encrypted ones [Kessler, 2010]. As another example, fairness-aware machine learning models intend to provide decisions independent of some given variables, especially those considered sensitive [Robert et al., 2020; Holstein et al., 2018]. As a result, sensitive personal attributes of the users are not incorporated in or inferred from these decision systems.

Our work focuses on the problem of privacy leakage in conversational texts, and proposes to alleviate it by detecting and rewriting the sensitive utterances. To pave the way towards our task, we first review the machine learning methods developed for privacy protection in Section 2.1. Some of these ideas will be served as guidelines for our design of privacy-aware text rewriting in Chapter 4. Then, we introduce the related research on personal information detection in Section 2.2, which is the background of Chapter 3. Finally, we review the recent progress of controllable text generation in Section 2.3 related to our privacy-aware rewriting research in Chapter 4 and Chapter 5.

2.1 Privacy Protection in Machine Learning

There are several techniques for privacy protection in machine learning, motivated by different settings. In this section, we will give a brief introduction of these works and connect them with our work.

2.1.1 Differential Privacy

Using data of users to train machine learning models is a common practice in many IT companies. However, these models may release private information about some users involved in the training dataset [Shokri et al., 2017]. Then, the challenge is *how to guarantee the user information in training dataset is not likely to be inferred by querying a machine learning model?*

Differential Privacy is a property that guarantees the effect on the model outputs is minor, as quantified by a parameter ϵ, given an arbitrary single element removed or added in a dataset. With such constraint, the query results cannot be used to infer the attributes about any single individual in the dataset, and therefore privacy is preserved.

ϵ-Differential Privacy provides an essential mathematical formulation for the property. Given a randomized algorithm, represented as a function $f(\cdot)$, if for all databases D_1 and D_2 differ on a single element, and all subsets $S \subseteq Im(f)$

$$Pr[f(D_1) \in S] \leq \exp(\epsilon) \cdot Pr[f(D_2) \in S]. \tag{2.1}$$

Then, the l_1 sensitivity Δf of function f is determined over all pairs of neighboring D_1 and D_2 as:

$$\Delta f = \max_{\substack{D_1, D_2 \\ \|D_1 - D_2\|_1 \leq 1}} \|f(D_1) - f(D_2)\|_1 \tag{2.2}$$

which captures the upper bound of the output changes given the modification of a dataset is less or equal to one.

Group Differential Privacy extends Differential privacy to k groups by replacing bounding parameter ϵ with $k \cdot \epsilon$ for the cases that several elements are sequentially modified, *e.g.*, for D_1 and D_2 differing on k items,

$$Pr[f(D_1) \in S] \leq \exp(k \cdot \epsilon) \cdot Pr[f(D_2) \in S]. \tag{2.3}$$

Generalized Differential Privacy for text. In NLP applications, the difference between two datasets could be estimated by a generalized distance $dist(D_1, D_2)$ [Fernandes et al., 2019], *e.g.,* Earth Mover Distance on word embedding [Kusner et al., 2015a],

$$Pr[f(D_1) \in S] \leq \exp(\epsilon \cdot dist(D_1, D_2)) \cdot Pr[f(D_2) \in S] \tag{2.4}$$

Differential privacy was proved effective in distinguishing writing style whilst preserving enough rest information and variation for accurate content classification [Fernandes et al., 2019].

Differential privacy is a strong constraint which applies on the whole database. Applying classical differential privacy to language models, with regard to token prediction, may lead to poor model performance [Shi et al., 2021]. In our work, we refer to the idea from differential privacy and integrate fairness objectives while training language generation models.

2.1.2 Algorithmic Fairness

Fairness is a long-standing topic in philosophy, psychology, and recently in machine learning research. In machine learning, an algorithm is *fair*, or have incorporated *fairness*, if the results of the models are independent of given variables. This property is used to make sure that the decision do not reflect discrimination towards particular groups of people [Mehrabi et al., 2021]. An example of an AI system with biased decisions is that African-American offenders are usually falsely predicted to be at a higher risk of committing a crime than Caucasian offenders.[1] We consider a language generation system is aware of privacy attributes, *e.g.*, gender and race, if the output utterances could not be discriminated to a certain group. The sensitive attributes of generated sentences could be protected by incorporating algorithmic fairness on those attributes. We introduce several definitions of fairness, which inspire our model design for privacy-aware text generation in Chapter 4.

- **Equal Opportunity**. "A binary predictor $\widehat{Y}$ satisfies equal opportunity with respect to A and Y if $P(\widehat{Y}=1|A=0,Y=1)=P(\widehat{Y}=1|A=1,Y=1)$" [Hardt et al., 2016]. This indicates that a person in a positive class should be of the same probability to be assigned positive outcome for both protected and unprotected groups, *e.g.*, gender and race. An algorithm is considered to be fair under equal opportunity if its true positive rate (TPR) is the same between different groups.
- **Equalized Odds**. "A predictor $\widehat{Y}$ satisfies equalized odds with respect to protected attribute A and outcome Y, if $\widehat{Y}$ and A are independent conditional on Y. $P(\widehat{Y}=1|A=0,Y=y)=P(\widehat{Y}=1|A=1,Y=y)$, $y\in\{0,1\}$" [Hardt et al., 2016]. This indicates that a person in both positive class and negative class should be of the positive output for both protected and unprotected groups. Equalized odds is similar to equal opportunity, but it considers false positive rate (FPR), in addition to TPR.
- **Demographic Parity**, also known as statistical parity. "A predictor $\widehat{Y}$ satisfies demographic parity if $P(\widehat{Y}|A=0)=P(\widehat{Y}|A=1)$" [Kusner et al., 2017; Dwork et al., 2012]. The person in each protected group has the same probability of being classified with the positive outcome.

Among these aforementioned methods, demographic parity does not depend on the confusion matrix, making it generalizable to privacy-aware language generation tasks. We will use fairness as additional constraints for obfuscating rewriting models, as discussed in Chapter 4.

2.1.3 Adversarial Training for Deep Neural Network

Recently developed deep learning models have achieved state-of-the-art performance in many tasks [Devlin et al., 2019a; Liu et al., 2021]. However, the prediction process

[1] https://www.theguardian.com/technology/2016/sep/08/artificial-intelligence-beauty-contest-doesnt-like-black-people

of these back-box models are complicated, therefore it is challenging to integrate rigorous fairness property to these models. Adversarial learning [Zhang et al., 2018a; Li et al., 2018b] exhibits a way toward more fair representations by mitigate the bias of the groups with sensitive attributes. The models are generally trying to maximize the accuracy of the predictor on Y, and simultaneously minimize the performance of adversary to predict the protected sensitive attributes. As demonstrated in Figure 2.1,

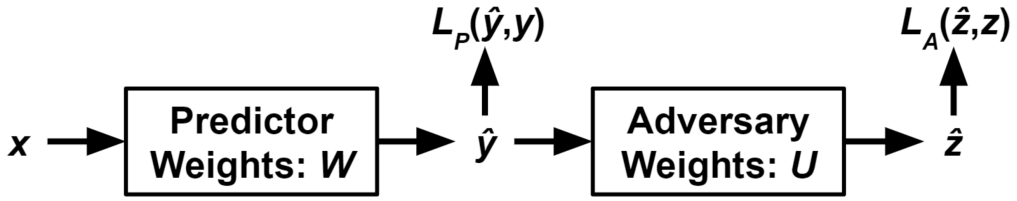


Figure 2.1: The architecture of the adversarial network, by [Zhang et al., 2018a].

the adversarial framework consists of two parts, namely predictor and adversary. The predictor is trained to predict the Y, given X, by minimizing $L_p(\widehat{y}, y)$. On the other hand, the output layer $\widehat{y}$ is passed to another adversary network, which attempts to predict the sensitive attribute Z. To achieve **Demographic Parity**, as an example, the predictor aims at preventing the adversary from learning Z. Given the various fairness definitions to be achieved, the adversaries may vary with different inputs and losses.

2.2 Personal Information Detection

In the last section, we introduced machine learning methods for eliminating the influence of specific attributes. In this section, we will connect our privacy leakage detection task to some relevant tasks. Authorship anonymization, in Section 2.2.1, tries to guarantee the individuals are unidentifiable based on their disclosed information, although the information itself could still be sensitive. Text anonymization, in Section 2.2.2, identifies and tags the sensitive concepts in pre-defined categories. The anonymization process harms the utility of the original sentence, while our privacy-aware rewriting attempts to trade off between privacy protection and utility of the output sentences. Our main approach to detect privacy leakage is based on *i)* natural language inference models trained on weakly supervised alignment data, in Section 2.2.3, and *ii)* regression models for scoring the sensitivity of the texts, in Section 2.2.4.

2.2.1 Authorship Anonymization

The target of authorship anonymization is to eliminate the trials of individual users from others, while their (sensitive) personal information is still collected and utilized

by service providers. The anonymization process could be formulated by data transformation or text rewriting to new ones with differential privacy (DP) property, as DP guarantees the personal trials and attributes could not be inferred back from the decision models. SynTf [Weggenmann and Kerschbaum, 2018] represents a series of research that protects the privacy of textual data, focusing on the numeric vector representation. The original feature vectors of a given document are replaced with or transformed to vectors that are differential private but with high utility on original target tasks. Embedding Reward Auto-Encoder (ER-AE) [Bo et al., 2021] further extends the problem to natural language and incorporates the differential privacy to text generation model through reinforcement learning by the rewards for *i)* removing personal traits and *ii)* semantic coherence and grammatical fluency.

Authorship anonymization considers the privacy is preserved if the user is not identifiable from a large group of users. The DP approaches are mostly mending the stylistic of the author but not the content, *i.e.*, semantic of the original text. Because DP is only guaranteed on the post-processed outputs of an analyzing system, your sensitive personal information in the original (input) texts is still under potential privacy risk, when those texts are disclosed to or misused by a third party. Our work focuses on detecting and eliminating the detailed sensitive personal information of the users, instead of eliminating their identifiability.

2.2.2 Text Anonymization

Data anonymization provides a more strict privacy protection paradigm, as it identifies and anonymizes the pre-defined sensitive information within given data samples [Medlock, 2006]. Our work refers to the data in form of text. The text anonymization basically conducts the following two steps:
Identification process detects the sensitive phrases, spans of tokens that refer to sensitive concepts, in the documents. For example, 'Joe Bloggs' and 'Somerton Bank' disclose the name and job of the person mentioned in the example. [Microsoft, 2021] summarizes some sensitive concepts as private, such as card numbers, names, locations, financial data and etc.
Anonymization process neutralizes the sensitive reference by removal, categorization and pseudonymization, as suggested by [Medlock, 2006]. They are replacing the sensitive reference with a placeholder '<REF>', a categorical label, or a variant expression of the same type, respectively. Table 2.1 gives examples for these anonymization methods.

removal:	Joe Bloggs works at Somerton Bank $\longrightarrow$ <REF> works at <REF>
categorisation:	Joe Bloggs works at Somerton Bank $\longrightarrow$ <PER> works at <ORG>
pseudonymisation:	Joe Bloggs works at Somerton Bank $\longrightarrow$ Phil Day works at Higgins Bank

Table 2.1: Examples of text anonymization processes, by [Medlock, 2006]

Due to the fact that text anonymization focus on sensitive phrases, it provides protection on fine-grained granularity in text. Presidio [Microsoft, 2021] is a representative text anonymization application which is used in businesses. However, the

main anonymization approach has two limitations. The first one is that it does not consider personal aspect of the users and social aspect of the texts. In particular, the sensitivity of the same text varies from different senders/receivers of the text and target of the conversation. For example, mentioning 'Bill Gates' is probably not considered sensitive in my conversations, as I do not know him personally, while it could be sensitive in the dialogues of Mr Gates' close relatives. Our detection module could fix this issue by defining personalized sensitive information for each user. The second limitation is that replacing the sensitive references could weaken the semantic relevance and grammatical fluency of the sentences. Our rewriting module explores the rewrites that trade off among multiple constraints.

2.2.3 Weakly Supervised Data Alignment

In our work, we assume that we have the access to some of the sensitive information by interlocutors. Identifying the leakage of personal information could be formulated as linking the utterances that infer corresponding personal information. The existing dataset [Zhang et al., 2018b] collected personal information descriptions and utterances from the same speaker. However, there is not utterance-level inference labels for supervised training. Weakly supervised data alignment [Hessel et al., 2019a] is used to discover the implicit alignments between two sets of utterances. In previous work, the alignment methods were explored for concept-word alignment [Lee et al., 2018] and image-sentence alignment in multi-modal documents [Hessel et al., 2019a]. Given two sets of data X and Y for alignment, $M = \{m_{ij}\}$ is the similarity matrix between all paired elements in X and Y, where $m_{ij} = sim(x_i, y_j)$. As better alignment indicates higher set similarity, the alignment problem is formulated as optimizing the weighted average of the similarity score

$$sim(X, Y) = \sum_{i,j} w_{ij} \cdot m_{ij}. \tag{2.5}$$

The different heuristics of the alignment algorithms are reflected by various constraints for optimising $sim(X, Y)$. Linear Sum Assignment Problem [Hessel et al., 2019a] assumes 0-1 hard alignment and sum of each row and column of the weight matrix W is less or equal than one,

$$\forall i, \sum_j w_{ij} \leq 1; \forall j, \sum_i w_{ij} \leq 1; \forall i, j, w_{i,j,k} \in \{0, 1\}. \tag{2.6}$$

Optimal Transport [Kusner et al., 2015a] optimises soft alignment and sum of each row and column of the weight matrix W is less or equal than one,

$$\forall i, \sum_j w_{ij} \leq 1; \forall j, \sum_i w_{ij} \leq 1; \forall i, j, w_{i,j,k} \in [0, 1]. \tag{2.7}$$

We observe that there is no guarantee of the number of the aligned elements in our personal information inference task. In Section 3.4.2, new alignment methods are

proposed based on our observation.

2.2.4 Text Classification and Regression

The measurement of sensitivity of the information included in a text is one of the essential parts to decide whether the text requires protection. It could be defined as binary classification problem, whether a given text is sensitive or not sensitive [Berardi et al., 2015]. Regression model could also be utilized to evaluate the sensitivity score of the text [Xu et al., 2021]. In this section, we give an overview of deep learning methods for text classification and regression.

Framework. Both text classification and regression models share similar framework with each other, as demonstrated in Figure 2.2. Given an input text x, the feature extraction model $f(\cdot)$ transform it into intermediate feature representation $h = f(x)$. We reuse the pre-trained large-scale language models as backbone feature extractor, as they have been demonstrated to achieve outstanding performance in many downstream NLP tasks [Qiu et al., 2020]. Then, an output layer $g(\cdot)$ transform the representation to the predicted output y of the task, *e.g.*, classification or regression. The model is trained by minimizing the loss $\mathcal{L}$ between model prediction y and ground truth label y'.

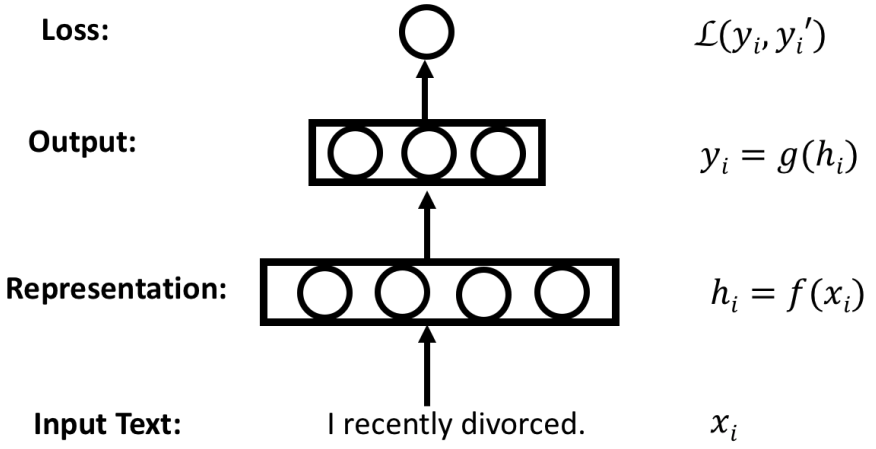


Figure 2.2: The framework of text classification or regression.

Classification and Regression. The main different adaptations to classification and regression are the output layers and loss functions. For regression problems, the output layer generally conduct a linear transformation $\mathbf{W}$ on h, *i.e.*, $y = \mathbf{W} \cdot h + b$, where b is a bias parameter. Mean Squared Error (MSE) serves as loss function for optimization, when training on corpus $\mathcal{D} = \{x_i, y_i'\}_{i=1}^{N}$ and N is the total number of training samples.

$$\mathcal{L}(\mathcal{D}) = \frac{1}{N}\sum_{i=1}^{N}(y_i - y_i')^2 \tag{2.8}$$

For classification problem, the output layer normalize the outputs to probability dis-

tribution using an additional softmax module, *i.e.*, $\mathbf{y} = \text{softmax}(\mathbf{W} \cdot h + \mathbf{b})$. Cross Entropy is used as loss function for optimizing the classifier.

$$\mathcal{L}(\mathcal{D}) = \frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} \mathbb{1}\{y'_i = c\} \log y_{i,c}, \tag{2.9}$$

where C is the total number of categories.

2.3 Privacy-Aware Text Generation

The second task to solve in this thesis is privacy-aware text rewriting. Privacy leakage in text can be mitigated by rewriting sensitive expressions into less or non-sensitive ones. The rewriting problems are often formulated as controllable natural language generation, which yields text according input control signals. We consider the personal information as control signals that guide the rewriting. A brief introduction of language generation model will be given in Section 2.3.1. Then, in Section 2.3.2, we describe the generation models that incorporate control signals, such as style attributes. Our rewriting model starts from this series of research and tackles more challenge settings, such as *i)* open-domain description of personal information and *ii)* multiple rewriting strategies.

2.3.1 Language Generation

Language generation task is to generate natural language output (text), based on the inputs, *e.g.* structured data, key words, images. Our work has a focus on natural language generation with text as inputs, which is similar to the settings of paraphrasing and translation [Bannard and Callison-Burch, 2005]. Given the input text X, the generation model predict the output word sequence Y with probability $P(Y|X)$.

$$P(Y|X) = \prod_{t=1}^{T} P(y_t|X, y_{<t}) \tag{2.10}$$

where Y is a sequence of T words $\langle y_1, \cdots, y_T \rangle$. The prediction of the tth word is based on both input X, and previously generated $t-1$ words.

$$P(y_t|X, y_{<t}) = P(y_t|X, y_1, y_2, \cdot, y_{t-1}) \tag{2.11}$$

The mainstream generation models are designed as encoder-decoder architecture, as demonstrated in Firuge 2.3. The encoder transform the discrete input text, *e.g.*, '*How are you?*', into an intermediate feature representations as a continuous vectors. Then, the decoder generate the output words sequentially, given the intermediate representations.

Model Architectures. Deep learning model for language generation could be summarised into two stages. In the first stage, recurrent neural network (RNN) are used

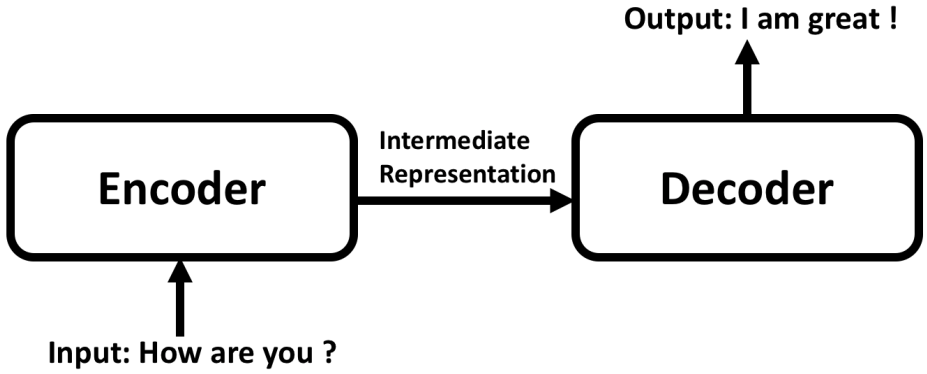


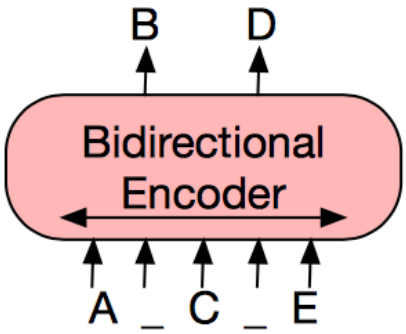
Figure 2.3: Encoder-decoder architecture for Language Generation. The encoder encodes the input sentence into an intermediate representation. Then, the decoder generates the output sentence based on the representation.

to capture the contextual information of words with short-term memory, such as GRU [Chung et al., 2014] and LSTM [Hochreiter and Schmidhuber, 1997]. In practice, bi-directional GRUs and LSTMs are used to incorporate information from both left and right sides of a word [Chiu and Nichols, 2016]. More recently, fully-connected self-attention models (*i.e.* Transformer) are proposed to capture global context of the words in a document [Vaswani et al., 2017]. Our rewriting models in Chapter 4 utilize the Transformer as backbone architecture for back-translation.

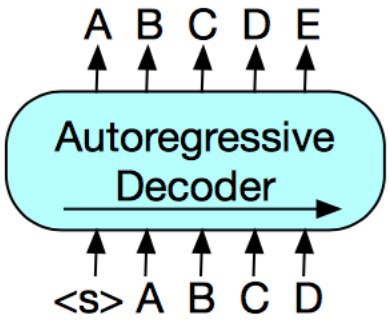
Pre-trained Generative Models. Deep learning based generation models normally have a huge number of parameters, while the training corpora for downstream tasks, *e.g.* our privacy-aware text rewriting, are relatively small. Pre-trained language models for generation is proved effective for many language generation tasks, such as machine translation, text summarization, and dialogue generation [Brown et al., 2020; Lewis et al., 2020]. These models are trained on a large-scale corpus and fine-tuned on downstream tasks. The ideas of pre-training methods for three representative models are illustrated in Figure 2.4. BERT [Devlin et al., 2019a] is pre-trained by predicting the masked tokens given their context. GPT [Radford et al., 2018] predicts tokens auto-regressively, namely predict the t-th token given previous $t-1$ tokens. BART [Lewis et al., 2020] takes the advantages of both BERT and GPT, by utilizing bidirectional encoder and autoregressive decoder. Our rewriting models in Chapter 5 are fine-tuned based on BART model.

2.3.2 Controllable Text Generation

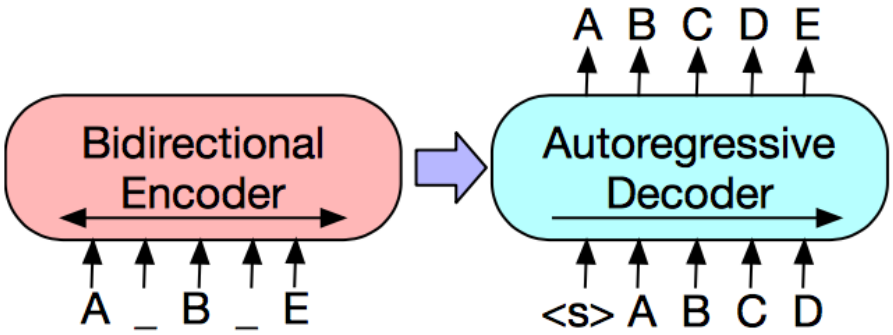
Incorporating specific attributes in the generated text, mostly defined as style transfer, recently attracts attention in NLG research. Different attributes, formality [Rao and Tetreault, 2018], politeness, authorship, simplicity [Xu et al., 2016a], sentiment [Fu et al., 2018] and etc, leads to various sub-tasks of controllable text generation.



(a) BERT [Devlin et al., 2019a] replace random tokens with masks and encodes the document bidirectionally. The prediction of missing tokens depends on the full context, therefore, using BERT for generation is not straightforward.



(b) GPT [Radford et al., 2018] predict tokens auto-regressively, which could be used for generation. However, it cannot learn bidirectional interactions, as all words can only condition on left context.



(c) BART [Lewis et al., 2020] takes the advantages of both BERT and GPT, as (1) bidirectional encoder incorporates both left and right context and (2) autoregressive decoder is adaptive for generation tasks.

Figure 2.4: A comparison of BART, with BERT and GPT, by [Lewis et al., 2020].

In general, the methodologies to solve the problem could be categorised based on whether the dataset possessed parallel text with different styles or several non-parallel corpora.

2.3.2.1 Methods on Parallel Data

Most style transfer model adopt encoder-decoder architecture for generation, but with additional style attribute S as input of the model $P(Y \mid X, S)$. Given the parallel corpus, style transfer model can be optimised via supervised training [Xu et al., 2012; Mathews et al., 2016; Rao and Tetreault, 2018]. The collection of parallel corpus for each sub-task is challenge, meaning that their scale should normally be smaller than those corpora for other generation tasks, such as machine translation [Koehn et al., 2005; Tiedemann, 2012] and text summarisation [Rush et al., 2015; Nallapati

et al., 2016]. [Niu et al., 2018] proposed to jointly learn style transfer model and machine translation model. The experiments showed that the auxiliary task significantly improved the performance of style transfer model.

2.3.2.2 Methods on Non-Parallel Data

It is difficult to obtain the parallel data for rewriting with a specific constraint. For example, rewriting a text into a paraphrase with the style of Mark Twain's novels is excessive for crowd-source workers. On the other hand, collecting documents with specific attributes is traceable, *e.g.*, a collection of Mark Twain's novels. Hence, the majority of related work investigates the approaches based on non-parallel corpora. Two representative series of work are *disentanglement* and *prototype editing*.

Disentanglement usually includes three steps, *i.e.*, *encode*, *manipulate* and *decode* [Li et al., 2018b; Prabhumoye et al., 2018a]:

1. Encode original text x into a latent representation z;
2. Manipulate z to remove the source attribute to z';
3. Decode z to text x' with target attribute a'.

One example is removing the personal attributes in text by encoding the text into continuous high-dimensional vector space, $z \in \mathbb{R}^k$, and the attributes are extracted and removed using adversarial methods [Li et al., 2018b; Elazar and Goldberg, 2018]. [Prabhumoye et al., 2018a] utilizes pivot language as latent representation, and translate x into pivot language, *e.g.*, English to French. Then, the attributes are transferred during back-translation from French to English.

Prototype editing follows a pipeline, *i.e.*, *delete*, *retrieve* and *generate* [Li et al., 2018a; Sudhakar et al., 2019; Madaan et al.], for attribute manipulation:

1. Detect and delete $\{w_a\}$, a set of the words that reflect the attributes in input sentences x, remaining content-only words in sentence $\hat{x} = x - \{w_a\}$;
2. Retrieve candidate text expressions c' carrying the desired attribute a';
3. Generate fluent sentences with new attribute based on $\hat{x}$ and c'.

In this thesis, *disentanglement* and *prototype editing* are explored and utilized in Section 4 and Section 5, respectively. The related work introduced in this sub-section is mostly steering the attribute of a text to another target attribute, while our work goes beyond *steering*. Our work in Section 4 proposes to protect the sensitive attribute by obscuring texts to neutral expressions. Our work in Section 5 explores three strategies for rewriting.

2.3.2.3 Methods on Low-Resource Data

In our open-domain privacy-aware text rewriting task, in Section 5, the personal information is very sparse and collecting the rewriting samples is expensive and time-consuming. Under low-resource setting, it is almost infeasible to train end-to-end models from scratch. The controllable generation system could be decomposed into two distinct components [Dathathri et al., 2019], such as

$$P(Y|X,S) = \frac{P(Y|X)P(S|Y,X)}{P(S|X)} \tag{2.12}$$

$$\propto P(Y|X)P(S|Y,X) \tag{2.13}$$

$$= P(Y|X)P(S|Y) \tag{2.14}$$

where X is the original text, Y is the generated text with style transfer, and S is the variable for style.

- *Language model* $P(Y|X)$ controls the linguistic quality of the generated content, such as grammatical fluency and semantic relevance.
- *Attribute controller* $P(S|Y)$ determines the contents relevant to the target attribute, *e.g.*, a scorer for selecting simpler words in text simplification task.

The main advantage of decomposing the generator is that both sub-modules could be acquired or trained separately. The language model module benefits from fine-tuning large-scale pre-trained language models, as described in Section 2.3.1. The attribute controller module could be trained as a classifier in a specific domain. The classifiers could be plugged in as a sampler [Su et al., 2018] or jointly optimized distribution [Dathathri et al., 2019]. This work inspires our approach to integrate knowledge constraints into decoder in Section 5.3.2.

2.4 Summary

The privacy protection task for text is related to a wide range of research topics. Firstly, in Section 2.1, we discussed several machine learning methods that inspires privacy protection, including differential privacy, algorithmic fairness and adversarial training. Then, in Section 2.2, we introduced related work for detecting privacy leakage in text. Finally, in Section 2.3.2, we reviewed methodologies related to text rewriting with control signals.

Chapter 3

Privacy Leakage Detection in Conversations

In Chapter 1, we introduced the private information leakage problem in conversations. In order to prevent the disclosure of this information, we propose to detect and report the utterances with potential risk of disclosing sensitive personal information. In this chapter, we describe our work on detecting privacy leakage in conversations. We first introduce the usage scenario the privacy leakage detection system in Section 3.1. Then, we formulate the detection problem as an alignment problem between utterance and personal information in Section 3.2. In Section 3.4, we suggest to train the inference model using weakly supervised alignment framework, and two new methods are proposed for better alignment performance. We collect a new dataset to test the performance of alignment models in Section 3.3. In our privacy leakage analysis, in Section 3.5, we test our detection model on both human-human and human-bot dialogues. The results show that more advanced dialogue models generally tend to leak more personal information and our best model manages to detect most of them. Finally, we demonstrate our privacy monitoring system which integrates to a real-world social media conversation platform in Section 3.7.

3.1 Introduction

Personal information can be dispersed through various types of media. In this work, we focus on natural language utterances in conversations articulated by digital personal assistants (PAs) or humans. The ways of controlling such textual information vary significantly w.r.t. platforms, PAs, user preferences, and social circles. Since there is no universally applicable control strategy, we take the first step towards privacy protection by designing a Personal Information Leakage Detection module (PILD) that *warns* users or *alerts* PAs whenever an utterance is associated with personal information, as illustrated in Figure 3.1. The warning module gives authorized users the capability to control information leakage from the start. Then, it is up to users and the design of PAs to decide how they deal with utterances leaking personal information. PAs will communicate with other interlocutors using secure or approved utterances.

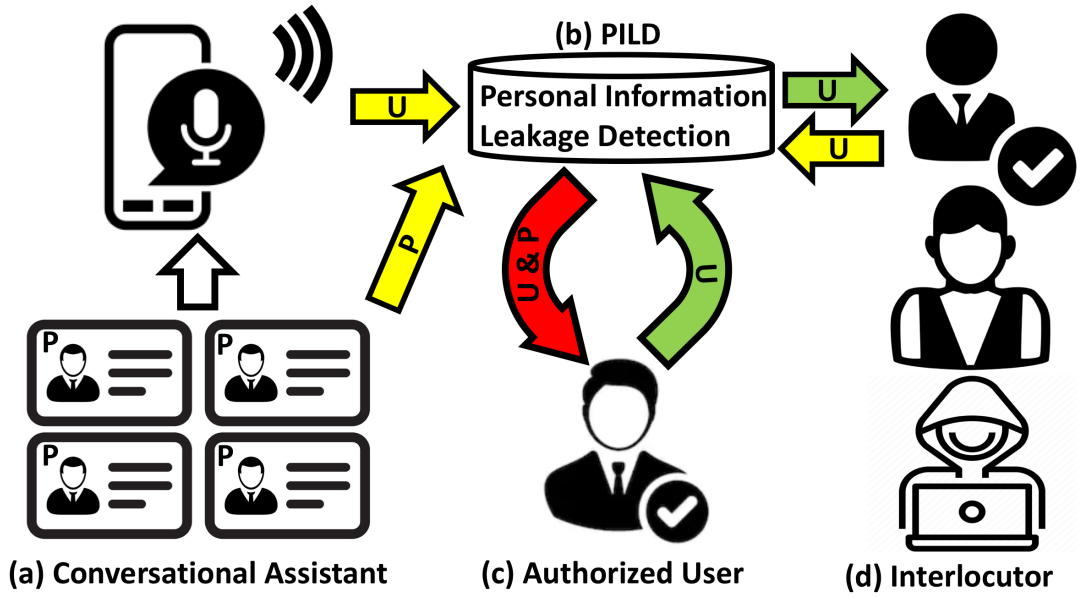


Figure 3.1: Given utterances (U) and personal information descriptions (P) from a conversational assistant (a), PILD module (b) detects risky utterances with corresponding personal information and sends a warning (red arrow) to an authorized user (c). The authorized user manually approve or reject the utterances. Then, only the approved utterances (green arrow) are sent to interlocutors (d) who could be authorized or malicious.

We formulate detection of utterances causing personal information leakage as a text alignment problem, which aims to link information leaking utterances to the corresponding textual descriptions of personal information. We consider personal information provided in text, because *i)* user profiles on popular social network platforms include a significant proportion of textual descriptions, and *ii)* it is natural for users to share their information with PAs in natural language. Figure 3.2 demonstrates an example of aligning utterances in a dialogue with a set of personal information descriptions. Those red lines depict the ground-truth alignments between utterances and personal information descriptions. The true alignments are sparse as not all utterances leak personal information, *e.g.*, U1, U3 and U6. Meanwhile, an utterance may be associated with more than one descriptions of personal information, *e.g.*, U2 and U4, and vice versa.

In the absence of direct supervision signals, we explore low annotation-cost solutions to this text alignment problem by considering a weakly supervised setting. In this setting, we only know *who speaks what* and what are the PI descriptions of each interlocutor during training, without knowing true alignments. The additional challenges are imposed by the complex relationships between utterances and descriptions of PI, which could be sparse alignment, and one-to-one, one-to-many, many-to-one, or many-to-many mapping.

The main contributions of the work in this chapter are the following:

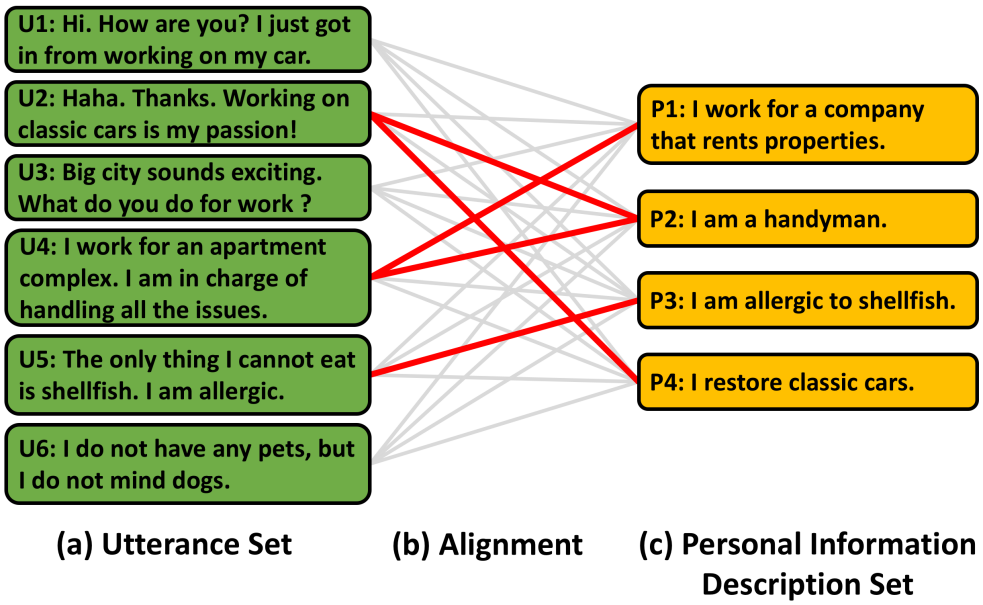


Figure 3.2: The alignment (b) of an utterance set (a) and a personal information description set (c) by a user. The matched sentence-level utterance-PI pairs are highlighted using red lines.

- We propose to protect privacy in conversation using PILD. Due to the lack of datasets for the new task, we construct a testing dataset PERSONA-LEAKAGE by extending the test set of the personalised dialogue corpus PERSONA [Zhang et al., 2018b] with alignment annotations through crowdsourcing.

- Under weakly supervised setting, we propose two novel alignment models Sharp-Max and Sparse-Max, which leverage coarse grained alignment signals to deliver sparse solutions. Our experiments on PERSONA-LEAKAGE show that our models achieve superior performance than competitive baselines.

- We empirically evaluated four representative dialogue models as persona assistants on PERSONA-LEAKAGE by letting them act as one of the interlocutors in a dialogue. We found that more advanced dialogue models are prone to leak higher proportion of personal information of the interlocutors they represent. Our PILD module works well on recently proposed dialogue agents.

- We design a privacy-leakage monitoring system that intervenes in risky responses and reports them to authorized users. To show the plausibility of our design, we develop and integrate the system to a real-world social media conversation platform.

3.2 Problem Statement

A dialogue between two interlocutors A and B is composed of two sets of utterances U_A and U_B. The corresponding persona profiles P_A and P_B are two sets of PI descriptions. A personalized dialogue dataset $\mathcal{D} = \{\langle U_i, P_i \rangle | i = 1, 2, \cdots, N\}$ consists of $\langle U_i, P_i \rangle$ associated with the same interlocutor i in a conversation, where $U_i = \{u_{i,j} | j = 1, 2, \cdots, n_i\}$ and $P_i = \{p_{i,k} | k = 1, 2, \cdots, m_i\}$. In the weakly supervised setting, a $\langle U_i, P_i \rangle$ from the 'same interlocutor' provides a set-level training signal for learning an alignment between the utterance set and the PI description set. An alignment is a set of links between an utterances set and an description set. This can also be viewed as identifying the edges of a bipartite graph between the two sets of vertices U_i and P_i. In the absence of alignment annotation during training, we relax the problem by learning alignment strength between $u_{i,j}$ and $p_{i,k}$ as an association score $a_{i,j,k}$, which constitute an association matrix $\mathbf{A}_i \in \mathbb{R}^{n_i \times m_i}$ for each $\langle U_i, P_i \rangle$. Then, it is up to the system design of a PA or the preference of an interlocutor to decide if an association score indicates that $p_{i,k}$ is leaked through $u_{i,j}$. For example, one can check if $a_{i,j,k}$ is above a pre-specified threshold.

3.3 PERSONA-LEAKAGE Dataset

In order to evaluate models under the weakly supervised setting, we constructed a dataset `PERSONA-LEAKAGE` as the test set by annotating the test set of the personalized dialogue corpus `PERSONA` [Zhang et al., 2018b]. In that corpus, each dialogue is conversed between two human interlocutors, where each interlocutor is characterized by three to five descriptions of PI. A description of PI describes one aspect of that person, *e.g.*, 'I am a handyman'. For each dialogue, we collected link candidates by pairing each utterance of a interlocutor to each description of his PI. As a result, we constructed a set of link candidates for each interlocutor in a dialogue. For each link candidate, we asked three annotators to judge if the utterance indicates the corresponding PI description. A candidate was considered as *aligned* if at least two annotators agreed on that decision. In total, we annotated alignments for 968 dialogues, in which there are 6,894 aligned utterance-PI pairs out of 67,601 candidate pairs.

Moreover, in order to understand the user perception on sensitivity of PI, we collected a set of all possible PI descriptions in test and dev set of `PERSONA`, and asked five annotators to judge if the descriptions were sensitive or not. A PI description is considered as sensitive if annotators would suggest not to share it with strangers, given that it describes their friends. We collected 306 descriptions (31.48% among all 972 descriptions) with more than 2 sensitive annotations.[1] For sensitivity score prediction in Section 3.6, we use the ratio of sensitive annotations for each description.

[1] Appendix A.1 describes more details about data collection.

3.4 Personal Information Leakage Detection

Recent advances in pre-trained language models, such as BERT [Devlin et al., 2019a], demonstrate their strengths of encoding semantic information into the produced text representations. Thus we apply a pre-trained language model $f(\cdot)$ (BERT in this work) to convert each utterance and each PI description into its representation vectors. As a widely accepted practice, we take the representation of the [CLS] token to represent an input text. Then, we apply a projection matrix $\mathbf{M}$ to map those vectors into a semantic space shared by utterances and PI descriptions,

$$\begin{aligned} \mathbf{r}_{i,j}^{(u)} &= \mathbf{M} \cdot f(u_{i,j}) \\ \mathbf{r}_{i,k}^{(p)} &= \mathbf{M} \cdot f(p_{i,k}) \end{aligned} \tag{3.1}$$

The association score between an utterance $u_{i,j}$ and a PI description $p_{i,k}$ is calculated by the cosine similarity between their representations, where $\langle \cdot, \cdot \rangle$ denotes the inner product of two vectors,

$$a_{i,j,k} = \frac{\langle \mathbf{r}_{i,j}^{(u)}, \mathbf{r}_{i,k}^{(p)} \rangle}{\|\mathbf{r}_{i,j}^{(u)}\| \|\mathbf{r}_{i,k}^{(p)}\|} \tag{3.2}$$

As we freeze the parameters of BERT in both training and testing, the only tunable parameters of this model is the matrix $\mathbf{M}$.

3.4.1 Alignment Framework

Learning an association matrix between an utterance set and a PI description set in the weakly supervised setting imposes two challenges. First, there is no ground-truth label to guide the alignment training. Second, an utterance may indicate zero, one, or multiple PI descriptions, while a PI description may also be associated with varying number of utterances.

Loss. To address the first challenge, we observe that *i)* a linked utterance-PI pair has high semantic relatedness; *ii)* the utterances in a dialogue are much more likely to correlate with the PI of its interlocutors than that of other interlocutors. The latter observation provides set-level alignment signals for contrastive learning. In light of this, we maximize the set-level aggregated associated scores for utterance-PI pairs from the same interlocutors $\langle U_i, P_i \rangle$, while minimizing those scores for the pairs from different interlocutors $\langle U_i, \widehat{P} \rangle$ and $\langle \widehat{U}, P_i \rangle$.

Alignment model. The second challenge imposes sparsity over the links in alignments. As it is difficult to enforce representation based cosine similarity values to approach zero, we introduce an alignment weight $w_{i,j,k}$ for each utterance-PI pair during training. The weight matrix $\mathbf{W}_i = \{w_{i,j,k}\}_{n_i \times m_i}$ puts a focus on the more reliable utterance-PI pairs and reduces the influence from irrelevant links. Then, the

similarity between U_i and P_i is the weighted sum of all elements in $\mathbf{A}_i$.

$$sim(U_i, P_i) = \mathbf{W}_i \odot \mathbf{A}_i = \sum_j \sum_k w_{i,j,k} a_{i,j,k} \tag{3.3}$$

where $\odot$ denotes hadamard product. High weights in $\mathbf{W}_i$ will enhance the corresponding association scores during training, while low weights or zeros in $\mathbf{W}_i$ discourage participation of those corresponding scores.

By putting two ideas together, the loss for the ith training sample is defined as:

$$\begin{aligned} \mathcal{L}(U_i, P_i) = &\max\{0, \alpha - sim(U_i, P_i) + sim(U_i, \widehat{P})\} + \\ &\max\{0, \alpha - sim(U_i, P_i) + sim(\widehat{U}, P_i)\} \end{aligned} \tag{3.4}$$

where $\widehat{U}$ and $\widehat{P}$ are randomly sampled from $\mathcal{D}$, α is a hyper-parameter controlling the margin of the loss. Then the loss on training set is the sum of all example losses $\mathcal{L}(\mathcal{D}) = \sum_{i=1}^{N} \mathcal{L}(U_i, P_i)$.

3.4.2 Sparse Alignment Models

The two models Sharp-Max and Sparse-Max differ in the regularizers used in $sim(U_i, P_i)$ for learning *sparse* weight matrices $\mathbf{W}_i$. The matrices $\mathbf{W}_i$ are expected to assign zeros or low weights to irrelevant pairs, while assigning high weights to the aligned pairs. They are formulated as a constrained optimization problem of the following form,

$$\begin{aligned} &sim(U_i, P_i) = \max_{\mathbf{W}_i} \{\mathbf{W}_i \odot \mathbf{A}_i + \gamma H(\mathbf{W}_i)\} \\ &s.t. \sum_j \sum_k w_{i,j,k} = 1, \forall j, k; w_{i,j,k} \in [0, 1] \end{aligned} \tag{3.5}$$

where $H(\cdot)$ is a regularization term that determines the sparsity of $\mathbf{W}_i$, and $\gamma \in \mathbb{R}^+$ adjusts the degree of regularization. If $\gamma \to 0$, the solution of the above problem is to assign the weight 1 to the maximal value in $\mathbf{A}_i$. As we expect more than one links in an alignment, the regularizer should encourage more non-zero entries in $\mathbf{W}_i$. If $\gamma \to +\infty$, the solution is weights with equal values, which aggregates $\mathbf{A}_i$ by averaging all association scores.

Sharp-Max utilizes entropy as the regularizer because uniform distribution achieves the maximum of entropy. In another words, this term encourages similar entries in $\mathbf{W}_i$.

Proposition 1. Let $\gamma \in \mathbb{R}^+$

$$H(\mathbf{W}_i) = -\sum_{j,k} w_{i,j,k} \log w_{i,j,k}$$

in Eq. (3.5), the solution of $\mathbf{W}_i$ is the following softmax function with temperature γ,

$$w_{i,j,k} = \frac{\exp(a_{i,j,k}/\gamma)}{\sum_j \sum_k \exp(a_{i,j,k}/\gamma)} \tag{3.6}$$

Proof Idea: The solution is derived by solving the Lagrangian of Eq. (3.5):

$$\begin{aligned}\mathcal{L}(\mathbf{W}_i, \lambda) = &\sum_j \sum_k w_{i,j,k} a_{i,j,k} \\ &- \gamma \sum_j \sum_k w_{i,j,k} \log w_{i,j,k} \\ &+ \lambda (1 - \sum_j \sum_k w_{i,j,k})\end{aligned} \tag{3.7}$$

Note that, when the temperature with $\gamma < 1$ is sufficiently small, the optimal $\mathbf{W}_i$ enlarges the differences of the values in $\mathbf{A}_i$ (Sharp-Max). If $\gamma = 1$, we got the conventional softmax, which is also referred to as **Soft-Max** in our experiments.

Sparse-Max considers the squared loss on $\mathbf{W}_i$ as the regularizer, as it controls the sparsity of the matrix by encouraging equal contributions.

Proposition 2. Let $\gamma = 1$,

$$H(\mathbf{W}_i) = -\frac{1}{2} \sum_{j,k} w_{i,j,k}^2$$

in Eq. (3.5), the solution of $\mathbf{W}_i$ is the *sparsemax* of $\mathbf{A}_i$ (Sparse-Max) [Martins and Astudillo, 2016].

$$w_{i,j,k} = [a_{i,j,k} - \tau(\mathbf{A}_i)]_+ \tag{3.8}$$

where $\tau(\cdot)$ is a dynamic threshold function and $[t]_+ = \max\{0, t\}$.

3.4.3 Experiments on Alignment Models

In this section, we explain our experiments for training decent alignment models. Our experiments basically show that our new alignment methods outperform information retrieval baselines and competitive alignment baselines.

3.4.3.1 Baselines

We apply the scoring function of two widely used information retrieval (IR) methods **TF-IDF** and **BM25** [Manning et al., 2008; Robertson and Zaragoza, 2009], and the most recent **BERT**-based IR [Dai and Callan, 2019] to measure the association between a PI description and an utterance.

We also consider the following competitive alignment models proposed in recent works.

- **MEAN** averages the contribution of association matrix, namely uniform weights $(1/(n_i \cdot m_i))$. We consider MEAN as the solution of a special case of our optimization problem with $\gamma \to +\infty$.

- **Avg-Max** [Lee et al., 2018] uses the average of the maximum similarity scores for all PI descriptions (**Avg-Max-P**) or utterances (**Avg-Max-U**).

- **LSAP** (Linear sum assignment problem) [Hessel et al., 2019a] optimizes hard alignments, where each row and column has less or equal than one link, *i.e.*, $\forall j, \sum_k w_{i,j,k} \leq 1; \forall k, \sum_j w_{i,j,k} \leq 1; \forall j,k, w_{i,j,k} \in \{0,1\}$.
- **OPT** (Optimal Transport) [Kusner et al., 2015a] optimizes soft alignments, where weights are in $[0,1]$ and sums of the weights on each column and row are less or equal to one, *i.e.*, $\forall j, \sum_k w_{i,j,k} \leq 1; \forall k, \sum_j w_{i,j,k} \leq 1; \forall j,k, w_{i,j,k} \in [0,1]$.

The weights of all alignment models are normalized to the sum of one.

3.4.3.2 Model Setting

In order to have a fair comparison, all alignment models share the same deep learning architecture which is composed of *i)* a pre-trained text representation model (BERT), *ii)* a learnable linear transformation layer, and *iii)* a weight computation module without back-propagation. The dimensions of pre-trained and final text representations are 768 and 256, respectively. We use Adam as optimizer for all experiments that require training. According to our preliminary experiments, we set learning rate to 0.01, batch size to 128 and train 200 epochs for all experiments.[2] We consider the hyper-parameters $\alpha \in \{0.0, 0.1, 0.2, 0.4, 0.8\}$ for all models and $\gamma \in \{1/4, 1/5, 1/6, 1/7, 1/8\}$ for Sharp-Max.

We evaluate the models by testing whether the alignment links between sets are correctly retrieved from all candidates links, following [Hessel et al., 2019a]. Given the ground-truth alignment between two sets, we evaluate the association matrix $\mathbf{A}_i$, by using precision at K (P@K)[3], R-Precision (Rprec), normalized discounted cumulative gain (NDCG) and mean average precision (MAP)[4]. In addition, we use Hellinger Distance (H-Dist) [Oosterhoff and van Zwet, 2012] $\frac{1}{N}\sum_i \frac{1}{2}\sum_{j,k}(\sqrt{w_{i,j,k}} - \sqrt{g_{i,j,k}})^2$ to quantify the matching rate of alignment weights $\mathbf{W}_i$ with ground-truth alignment weights $\mathbf{G}_i = \{g_{i,j,k}\}_{n_i \times m_i}$, where $g_{i,j,k}$ is normalized over j,k to sum to one.

3.4.3.3 Experimental Results

We compare our alignment models, Sharp-Max and Sparse-Max, with IR baselines and alignment baselines, in Table 3.1The proposed model consistently outperform baseline methods, indicating the effectiveness of our methods. H-dist is strongly correlated to other metrics, because better alignments lead to better H-dist. IR models significantly outperform random guess, showing that semantic information provided in utterances and descriptions provides strong guidance on inference. Although the naive MEAN does not enforce sparsity during training, it outperforms the unsupervised IR models with a large margin, more than 10% for all scores, showing that coarse grain signal is effective for learning semantic relevant for the PI leakage.

[2] We have explored learning rate in $\mathcal{R} = \{0.1, 0.01, 0.001\}$ and number of training epochs in $\mathcal{E} = \{25, 50, 100, 200, 400\}$.

[3] As the average and maximum number of alignment links are 3.56 and 9 in our corpus, we choose $K \in \{1, 3, 5\}$.

[4] https://trec.nist.gov/trec_eval/

Avg-Max, OPT and LSAP further outperform MEAN with a margin more than 2% for most of the metrics, as they apply the sparsity constraints in order to focus on aligned utterances and PI descriptions during training. Although these approaches set up competitive baselines on our task, SHARP-MAX and SPARSE-MAX achieve consistent improvement on all evaluation metrics. As SPARSE-MAX cuts off the weights of irrelevant pairs, it performs the best.

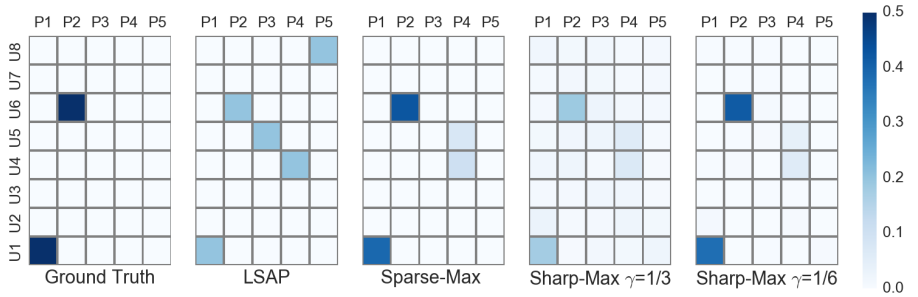
Model	P@1	P@3	P@5	Rprec	NDCG	MAP	H-Dist
RANDOM	0.1124	0.1050	0.1099	0.1107	0.4349	0.1919	N/A
TF-IDF	0.6716	0.5434	0.4294	0.5088	0.7548	0.5832	N/A
BM25	0.6824	0.5364	0.4207	0.4988	0.7535	0.5785	N/A
BERT	0.5923	0.4149	0.3257	0.3762	0.6789	0.4677	N/A
MEAN ($\alpha = 0.1$)	0.7573	0.6361	0.5230	0.6178	0.8331	0.7097	0.6801
Avg-Max-P ($\alpha = 0.4$)	0.7856	0.6748	0.5545	0.6566	0.8561	0.7486	0.3797
Avg-Max-U ($\alpha = 0.2$)	0.7785	0.6647	0.5452	0.6467	0.8493	0.7369	0.4680
OPT ($\alpha = 0.2$)	0.7725	0.6605	0.5448	0.6434	0.8470	0.7340	0.4822
LSAP ($\alpha = 0.4$)	0.7780	0.6670	0.5495	0.6522	0.8529	0.7434	0.4084
SOFT-MAX ($\alpha = 0.1$)	0.7676	0.6554	0.5341	0.6350	0.8421	0.7247	0.6042
SHARP-MAX ($\alpha = 0.4$, $\gamma = 1/6$)	0.7942	0.6763	0.5517	0.6618	0.8577	0.7499	0.3208
SPARSE-MAX ($\alpha = 0.4$)	**0.7970**	**0.6839**	**0.5597**	**0.6695**	**0.8612**	**0.7562**	**0.3032**

Table 3.1: Experimental results of random guess (RANDOM), unsupervised IR models (TF-IDF, BM25, and BERT), baseline alignment models (MEAN, Avg-Max-U, Avg-Max-P, OPT and LSAP), and our proposed models (Soft-Max, Sparse-Max and Sharp-Max).

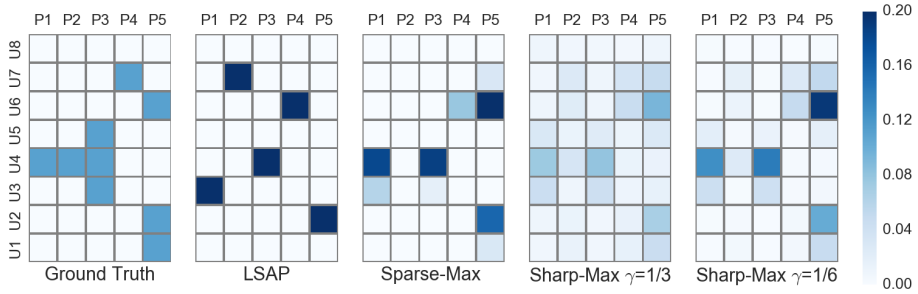
3.4.4 Analysis on Alignment Model

We visualize the association scores of each alignment model in Figure 3.3, in order to qualitatively demonstrate the strengths of our models. LSAP attempts to assign a fixed number of aligned pairs, *i.e.*, $\min\{n_i, m_i\}$, which will lead to unavoidable false positive alignment for sparse cases (U8-P5, U5-P3 and U4-P4, in Figure 3.3a LSAP) and false negative alignment for dense cases (U4-P1 and U4-P2, in Figure 3.3b LSAP). Avg-Max-P and Avg-Max-U also hold the similar drawback as the number of aligned pairs is exact the number of columns or rows, while does not depend on the cases. In contrast, SPARSE-MAX and SHARP-MAX manage to adapt the number of 'aligned pairs' (deep colored), therefore achieve alignments closer to the ground truth. For SHARP-MAX, we can adjust the sharpness of the weight matrix using sharpness parameter γ. Using sharper model with lower γ manages to alleviate the influence of the pairs with relatively low similarity scores. For SPARSE-MAX, more deterministic alignments are achieved by cutting off pairs with low association scores. Although SHARP-MAX and SPARSE-MAX do not differ much in terms of empirical performance, they are driven by different theories of regularization. The comparison between these two solutions proposed by us helps draw a conclusion that the similarity function should be designed to find a proper degree of sparsity, which does not depend much on a

particular choice of regularizer.



(a) A case with sparse alignment.



(b) A case with dense alignment.

Figure 3.3: Comparison of weights assigned to candidates between utterances (U1-U8) and personal information descriptions (P1-P5). (a) case 12 and (b) case 85 are test cases with sparse and dense alignments, respectively. The alignment weights of Ground Truth and LSAP are all normalized to the sum of one for each case.

3.5 Privacy Leakage Analysis in Chatbots

In order to understand the risk of privacy leakage in personalized chatbot, we collect and analyze human-bot dialogues using SOTA personalized chatbots and their competitors:

- $\mathcal{P}^2$ **Bot** [Liu et al., 2020] achieved SOTA performance on automatic metrics by incorporating mutual persona perception. $\mathcal{P}^2$ **Bot (w/ Persona)** and $\mathcal{P}^2$ **Bot (w/o Persona)** are models with and without personal information as input when generating responses.
- **Lost-In-Conversation** [Dinan et al., 2019] topped the human evaluations in ConvAI2 by fine-tuning a pre-trained language model GPT.
- **Seq2Seq-Attn** [Zhang et al., 2018b] is an LSTM-based sequence-to-sequence model incorporateing persona via an attention module.
- **Language Model** [Zhang et al., 2018b] is an LSTM-based language module for dialogue.

For each chatbot, we provided interlocutor A's dialogue history as input and the bots responded as interlocutor B. The personalized chatbots [Dinan et al., 2019; Zhang et al., 2018b] utilized B's personal information as auxiliary inputs for better generative results. We performed 60 dialogues and collected 770 utterances for each chatbot. The responses by those chatbots are analyzed in three dimensions.

- Personal Information Engagement (**PIE**) is the proportion of the utterances leaking PI,
$$|\text{Utterances have PI Leakage}|/|\text{All Utterances}|$$
- Disclosed PI Sensitivity (**DPS**) is the ratio of sensitive PI descriptions to the leaked ones,
$$\frac{|\text{Sensitive Disclosed PI descriptions}|}{|\text{Disclosed PI descriptions}|}$$
.
- Hits-at-K (**Hits@K**) is the percentile of the leaked PI that can be retrieved from top $K = 5/10$ results using alignment models.

Perplexity (**PPL**) and uni-gram **F1** are supplementary metrics that reflect the performance of bots [Liu et al., 2020].

Model	PIE	DPS	Hits@5/10	PPL ↓	F1 ↑
Language Model	02.13	06.45	29.03 / 32.26	51.61	13.59
Seq2Seq-Attn	04.39	06.54	18.64 / 22.03	39.54	15.52
$\mathcal{P}^2$ **Bot (w/o Persona)**	08.94	10.77	51.54 / 56.15	-	17.77
Lost-In-Conversation	14.68	09.39	79.34 / 82.63	-	16.83
$\mathcal{P}^2$ **Bot (w/ Persona)**	37.19	16.86	73.62 / 77.04	18.89	19.08
Human	43.83	27.75	55.07 / 66.52	-	-

Table 3.2: Analysis on the responses of personalized chatbots and human interlocutors.

We analyze the engagement and sensitivity of chatbots in human-bot conversations. The experiments are designed to show the risk of privacy leakage when using current chatbot models. For all generated utterances, we retrieved top 10 relevant PI using Sparse-Max. Then we asked annotators to select the leaked ones from the retrieved PI descriptions. Three annotators are asked to indicate if the utterances leak those PI descriptions. Majority voting is used to decide if the utterance is leaking corresponding personal information. The results are summarized in Table 3.2. Compared with bots *without PI* as inputs, such as **Language Model** and $\mathcal{P}^2$ **Bot (w/o Persona)**, the bots *with PI* as input, namely **Lost-In-Conversation** and $\mathcal{P}^2$ **Bot (w/ Persona)**, tend to acquire higher PIE with significantly higher magnitude. PIE of $\mathcal{P}^2$ **Bot** even approaches that score of human interlocutors. DPS is correlated to PIE showing that bots with higher PIE generally disclose higher portions of sensitive PI.

Although higher PIE and DPS for the chatbots *with PI* as input is expected, there is also a significant proportion of leakage for the bots without PI as input, *e.g.*, $\mathcal{P}^2$ **Bot (w/o Persona)**. This raises serious privacy concerns in future research on PAs.

Furthermore, Hit@K measures the ability of our system for detecting PI leakage. As a warning module, our model SPARSE-MAX manages to detect most of the utterances leaking PI[5]. Our system achieves around 80% of Hit@10 on the responses generated by the two most recent and advanced chatbots, **Lost-In-Conversation** and $\mathcal{P}^2$ **Bot (w/ Persona)**.

3.6 Sensitivity Scoring

We consider predicting the PI sensitivity scoring as a regression problem. We define the sensitivity score to be the ratio of annotators consider the personal information is sensitive as described in Section 3.3. We construct the model based on a pre-trained text representation model and a linear transformation layer. Mean squared loss is used as loss function during training. The experimental results are evaluated by Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). We compare a few models for text representations, such as averaged word vectors **Word2Vec** [Mikolov et al., 2013], **BERT** [Devlin et al., 2019a], and **ALBERT** [Lan et al., 2019]. For all models, we conduct five experiments and report the averaged scores and variances in Table 3.3. Deep learning models, BERT and ALBERT, are competitive and outperform Word2Vec. ALBERT works slightly better than BERT and achieves the best results.

Model	RMSE	MAE
Word2Vec	1.303 ± 0.011	1.039 ± 0.011
BERT	1.249 ± 0.031	0.996 ± 0.026
ALBERT	$\mathbf{1.232} \pm 0.033$	$\mathbf{0.987} \pm 0.033$

Table 3.3: The experimental results of linear regression models for sensitivity score estimation, using Word2Vec, BERT and ALBERT for sentence representations.

3.7 Application: Privacy Monitoring Service

We developed a privacy-leakage monitoring system on a social media conversation platform, which is Facebook Messenger in this work. This system detects an outgoing message imposing privacy risk and alerts authorized users to inspect it. Then it is up to the users to decide how they handle the message. Figure 3.4.a shows how conversations are conducted between interlocutors through a conversation platform. The outside interlocutors, on the right-hand side, may intentionally acquire sensitive information from the interlocutors requiring protection, *e.g.*, vulnerable users

[5] According to our preliminary experiments, SPARSE-MAX achieves the best Hits on the whole test set of `PERSONA-LEAKAGE`.

and digital personal assistants, by conversing with them on the conversation platform. Figure 3.4.b demonstrates the workflow of our privacy monitoring service. A privacy monitor detects the utterances that leak sensitive personal information and alerts the authorized users, *i.e.*, guardians of the vulnerable users or owners of the digital personal assistants. The authorized users can choose to modify the utterances into less sensitive ones or directly approve the original responses. The modified or approved utterances will be sent back to conversation platform and finally come forth to the outside interlocutors.

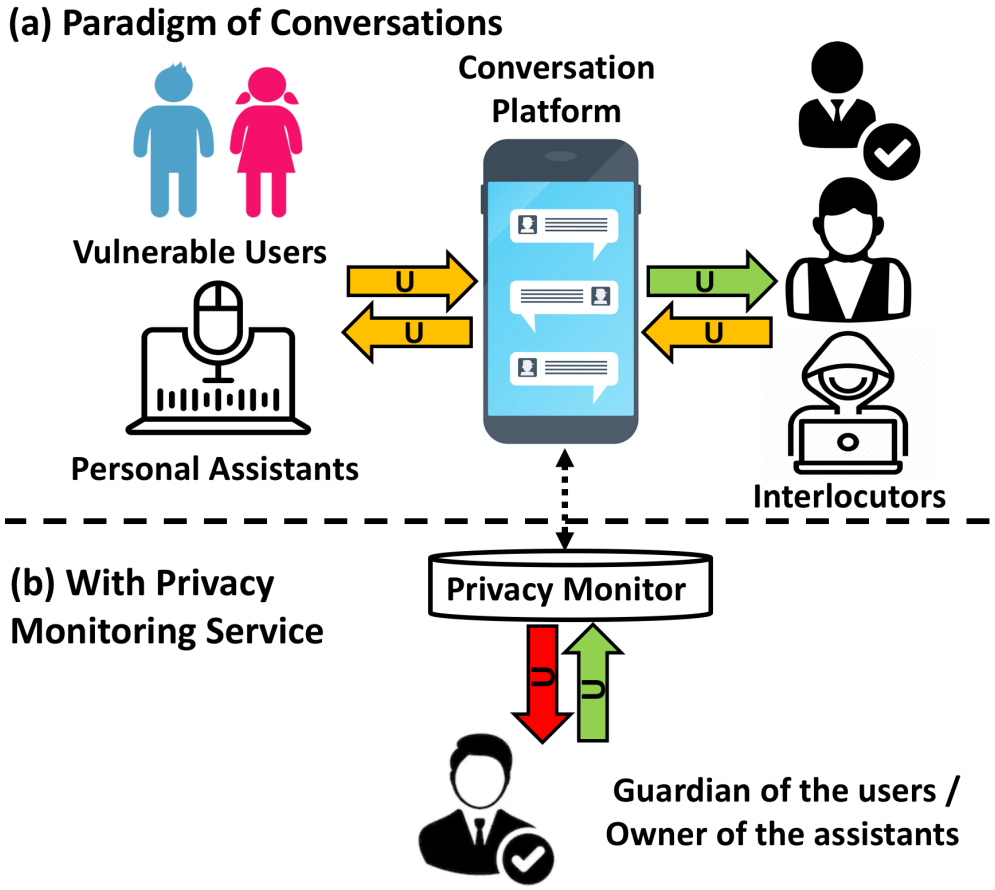


Figure 3.4: The paradigm of conversations with our privacy monitoring service. Interlocutors from outside (right) might get access to vulnerable users or your personal assistants (left) through a conversation platform. Some of the intended responding utterances (U) might be sensitive, as demonstrated in orange arrows. The privacy monitor is designed to detect and intercept the sensitive utterances in red. Only the authorized users are allowed to approve or to modify these messages. Finally, the approved or modified messages are sent to the outside interlocutors.

Our system supports two modes, designed for human-human and human-bot conversations, respectively. For human-human conversations, vulnerable users can only contact the unreliable interlocutors through our monitor. This paradigm could

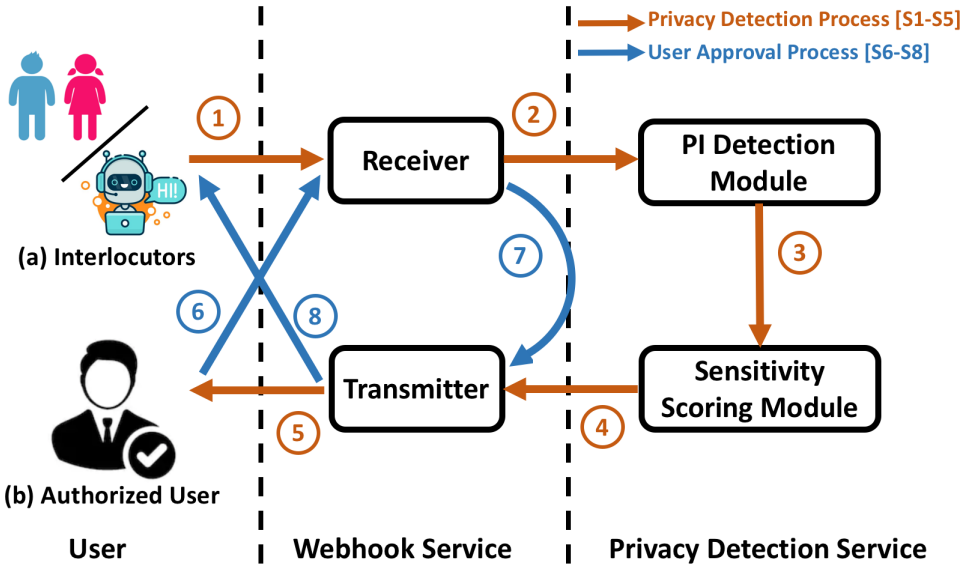


Figure 3.5: The system design and workflow of Privacy Monitoring Service (PMS). Each intended utterance from a vulnerable interlocutor passes a privacy detection process [S1-S5] and a user approval process [S6-S8], before it is sent to outside interlocutors.

be used in social media platforms for vulnerable group privacy protection. For human-bot conversations, we integrate a personalized chatbot as a personal assistant monitored by our system.

3.7.1 System Overview

In this section, we detail the system design and implementation of our Privacy Monitoring Services (PMS), as depicted in Figure 3.5, which contains two main components, a *Privacy Detection Service* and a *Webhook Service*.

Privacy Detection Service detects the utterances that disclose privacy by *i)* detecting relevant personal information, and *ii)* scoring the sensitivity of the retrieved personal information. Personal Information (PI) Detection module utilizes an unsupervised utterance-persona alignment model as described in Section 3.4, while sensitivity scoring module relies on a regression model as an estimator as described in Section 3.6. Given the PI detection and sensitivity scoring models, we deploy them as web service based on Flask[6] and exposes REST APIs to our Webhook Service. At each request, Privacy Detection Service receives the conversation context and the anticipated respondent utterance (S1 and S2). For the privacy leakage utterance, an alarm with the disclosed personal information descriptions and corresponding sensitivity scores are reported to an authorized user for approval (S4 and S5). The approved or written utterances are then sent back to outside interlocutors (S6 to S8).

[6]https://flask.palletsprojects.com/

3.7.2 Demonstration and Scenarios

We demonstrate a practical conversational scenarios in Messenger with our privacy monitoring service integrated. The interfaces for interlocutors and authorized users are demonstrated in Figure 3.6. For a sensitive message, the authorized user receives a warning with corresponding conversation history, intended responses and disclosed personal information. The message is blocked from displaying to the the outside interlocutor, until the authorized user approve it by responding a '[Heart]' or rewrite it by '[Reply]' to the alarm message.

We demonstrate our privacy monitoring service for human-human and human-bot conversations. In both scenarios, normal conversations interlocutors communicate without interruption, except privacy leakage parts are intercepted and notified to the authorized user for approval. A screenshot video of our demo system is available at Google Drive[7].

- **Human-human conversation.** Malicious outside interlocutors might get access to private information by communicating with vulnerable users, such as children or intellectually challenged people. PMS monitors the messages from the vulnerable users on behalf of their guardians.
- **Human-bot conversation.** Malicious outside interlocutors might obtain the personal information of a person by querying his digital personal assistants. PMS intercepts and reports the privacy leaking messages from digital personal assistants to the owners of the bots before the messages are sent out.

3.8 Summary

In this chapter, we have investigated the first problem towards privacy protection in conversations by detecting the utterances with the risk of privacy leakage. We proposed to learn the models via weakly supervised alignment problem and demonstrate the effectiveness of our approach. We collected `PERSONA` as test dataset and analyze the privacy leakage risk in dialogues. Finally, a privacy monitoring system was developed to show the feasibility of privacy leakage detection in real-world conversational platforms.

[7] https://drive.google.com/file/d/131Zq94FH1gi1EITqesnz9cGVsbdR7ttF/view

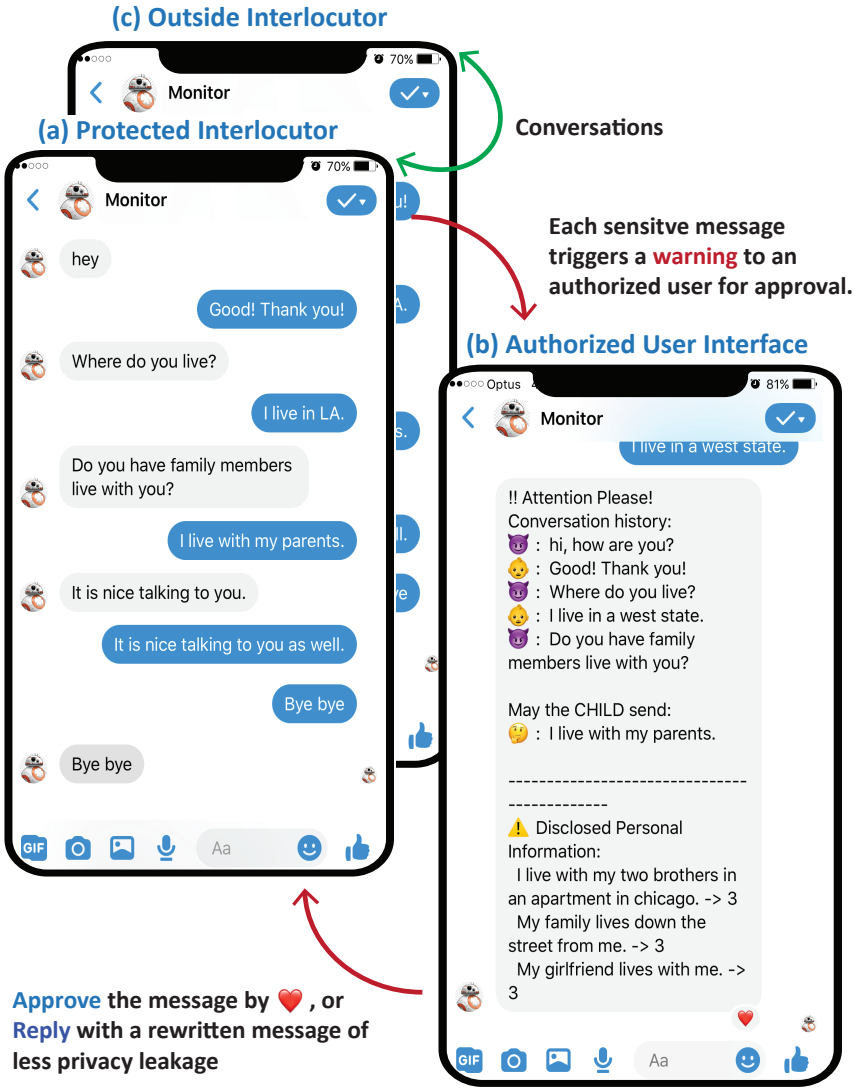


Figure 3.6: The screenshots of (a) an Interlocutor Interface chatting with (c) an outside interlocutor, and (b) an Authorized User Interface, with a warning message from (a) to (b) and an approved utterance from (b) to (a).

Chapter 4

Obscuring Personal Attributes for Privacy-Aware Text Rewriting

In Chapter 3, we presented our work on detecting utterances with privacy leakage risk, and demonstrated a monitor system for privacy leakage. Given the detected utterances with privacy leakage risk, the next challenge is to rewrite them into less sensitive ones. In Chapter 4 and Chapter 5, we investigate automatic privacy-aware text rewriting using deep learning based language generation models. Inspired by the related work of eliminating bias in text [Emmery et al., 2018; Strengers et al., 2020], we initialize our research by obscuring the personal attributes, *e.g.*, gender, political slant and race, in the original utterances. We follow back-translation framework [Prabhumoye et al., 2018b] for text style transfer without parallel training corpus in Section 4.2. Then, we explore how to obscure attributes in back-translation and propose to use *i)* adversarial training and *ii)* fairness-risk measurement to learn the obscuring strategy for rewriting, with more details in Section 4.3 and Section 4.4. We consider three datasets with gender, political slant and race as sensitive personal attributes of the writers, with more details in Section 4.5.1. Our experiments on these three datasets suggests that our rewriting methods manage to obscure the implicit sensitive attributes with different strength, while better privacy protection is normally accompanied with more loss of semantic relevance and grammatical fluency.

4.1 Problem Statement

Privacy-aware rewriting modifies text to obscure a sensitive attribute. The methodologies aim to minimize the loss of fluency as well as the change in the underlying semantics. We consider a setup in which we have a set of input text $\{X_1, ..., X_N\}$, where each text X_i is a word sequence $\langle x_1, ..., x_l \rangle$. Each text is associated with a sensitive attribute S, such as gender or race. The goal is to find a privacy-aware translator $f(X) : X \rightarrow Y$ to modify X into another word sequence $Y = \langle y_1, ..., y_l \rangle$, such that an attacker $g(Y) : Y \rightarrow S$ fails to predict the values of the sensitive attribute S from the translated text Y.

4.2 Privacy-Aware Back-Translation

Privacy-aware rewriting can be regarded as a special monolingual machine translation (MT) task, which aims to remove sensitive information through rephrasing. In our experiment, there is no existing parallel corpus to learn the patterns of privacy-preserved rewriting. We use Back-Translation to obtain a meaning-preserving representation in the target language, and translate the sentences back to the source language [Prabhumoye et al., 2018a]. Since we aim to preserve sensitive information, we consider the risk from an attacker in the back-translation phase. Directly using style transfer generation approach [Prabhumoye et al., 2018a] could be viewed as a solution for steering in Chapter 5, while the work in this chapter is a method for obscuring, which avoids providing false/deceptive information [Xu and Zhao, 2012].

In our work, the source language is English and the target language is French. Let Z denote the space of target language, we build two translation models $\mathcal{T}_{\text{en}\to\text{fr}} : X \to Z$ and $\mathcal{T}_{\text{fr}\to\text{en}} : Z \to X$, respectively. We use the Transformer-based model [Vaswani et al., 2017] for each translation model. The back-translation procedure is formulated as,

$$f(X) = \mathcal{T}_{fr\to en}(\mathcal{T}_{en\to fr}(X)) \tag{4.1}$$

For each input text, the outcome of this model is a sequence of words in English.

The goal of learning privacy-aware back-translation is two-fold. Firstly, it aims to find an optimal predictor f^* that minimizes an expected reconstruction loss

$$\mathbb{E}_{X,Y}[\mathcal{L}(f(X), Y)]$$

with $\mathcal{L}(f(X), Y) : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, which measures the discrepancy between predicted sequences $f(X)$ and true target sequences Y. Secondly, the predictor should be reasonably fair to S by achieving a low risk loss with regard to privacy $\mathcal{R}(X, Y, S) : \mathcal{X} \times \mathcal{Y} \times \mathcal{S} \to \mathbb{R}$. Let $\mathcal{F}$ denote the space of all possible predictors, we find the optimal rewriting model f^* by

$$f^* = \arg\min_{f\in\mathcal{F}} \mathbb{E}_{X,Y}[\mathcal{L}(f(X), Y)] + \alpha\mathcal{R}(X, Y, S) \tag{4.2}$$

where α controls the degree of privacy protection.

4.3 Adversarial Training

Given an accurate classifier, the risk of privacy is able to be estimated by the negative classification loss on the sensitive information. Our target is finding the representations that are good at reconstructing the sentences, while poor in predicting sensitive labels. The setting is well-aligned with generative adversarial networks [Goodfellow et al., 2014]. We construct the back-translation model as $f(X) = m(h(X))$, where $h(X)$ employs the two translators to map X into a sequence of hidden representations of decoded words in the source language. Then, $m(\cdot)$ maps the hidden representa-

tions into the corresponding words. An adversarial classifier $adv(h(X))$ is a linear classifier, which takes the mean of all hidden representations from $h(X)$ to predict S. The risk is formulated as adversarial classification loss $\mathcal{L}_c(adv(h(X)), S)$. The encoder $h(\cdot)$ is trained to fool the adversarial classifier $adv(\cdot)$ while optimizing the back-translation predication $f(X)$ in Eq.(4.4). Eq.(4.3) merely optimizes the adversarial classifier. The training is conducted by jointly optimizing the following two objectives:

$$\arg\min_{adv} \mathcal{L}_c(adv(h(X)), S) \tag{4.3}$$

$$\arg\min_{h,m} \mathcal{L}_g(m(h(X)), X) - \alpha \mathcal{L}_c(adv(h(X)), S) \tag{4.4}$$

where $\mathcal{L}_g$ is the cross entropy loss with Label Smoothing [Szegedy et al., 2016] for the transformer-based generator and L_c is the cross entropy loss for the adversarial classifier. The negative parameter $-\alpha$ is implemented by a gradient-reversal layer (GRL)[Ganin and Lempitsky, 2015] during back-propagation and α controls the intensity of adversarial training.

4.4 Fairness-Risk Measurement

In this section, we define the privacy risk loss using fairness risk measurement. The perfect fairness for rewriting is a statement of conditional independence of generated text $Y \perp\!\!\!\perp S|X$. Holding such condition, the sensitive translator conduct similar generation results. Therefore, attackers will not be able to infer the dependent attributes. A *privacy-aware* translator $f(X)$ learns a distribution $P(Y|X)$, while $P(Y|X, S = a)$ denotes the distribution of a *subgroup* translator depending on a particular demographic group attribute S. The conditional independence is formulated as,

$$P(Y|X) = P(Y|X, S = a) \tag{4.5}$$

[Agarwal et al., 2018] pointed out that given finite samples in training data, it is impossible to ensure perfect fairness on the test sample. An approximate formalism of fairness measurement is used to quantify the discrepancy of demographic parities, namely maximal deviation between subgroup predictions (MDSP) [Calmon et al., 2017].

$$\sup_{y,s,s'} |Pr(\hat{Y} = y|S = s) - Pr(\hat{Y} = y|S = s')| \tag{4.6}$$

where $\hat{Y}$ is a single variable.

Inspired by the single-variable MDSP, we define the sequential MDSP (SMDSP) for text rewriting as,

$$\sup_{a \in S} |\log P(Y|X) - \log P(Y|X, S = a)| \tag{4.7}$$

where Y is the generated sequences. We obfuscate the sensitive attribute by reducing

the discrepancy between privacy-aware translator and the most different subgroup translator.

The challenge of using the SMDSP is that it is optimized on the whole sequences. However, the state-of-the-art encoder-decoder architecture [Vaswani et al., 2017; Klein et al., 2017] generate words in a word-by-word manner.We derive an upper bound of SMDSP by applying calculus on the sequential deviation

$$\begin{aligned} D(X, Y, S = a) &\doteq |\log P(Y|X) - \log P(Y|X, S = a)| \\ &= |\sum_{i=1}^{l} \log P(y_i|X, y_{<i}) - \sum_{i=1}^{l} \log P(y_i|X, y_{<i}, S = a)| \\ &\leq \sum_{i=1}^{l} |\log P(y_i|X, y_{<i}) - \log P(y_i|X, y_{<i}, S = a)| \\ &\doteq \mathcal{U}_a(X, Y) \end{aligned}$$

The composition of MDSP for each word is an upper bound of SMDSP.

$$\mathcal{R}_u(X, Y, S) = \sup_{a \in S} \mathcal{U}_a(X, Y) \tag{4.8}$$

We replace the approximate fairness risk by its upper bound Eq.(4.8) and obtain a joint training objective.

$$\mathcal{L}_\alpha(X) = \mathcal{L}(f(X), X) + \alpha \mathcal{R}_u(X, Y, S) \tag{4.9}$$

In training, each subgroup translator is pre-trained beforehand with the training data labeled with the corresponding sensitive attribute value. Their parameters are kept fixed when minimizing the privacy-aware rewriting model.

4.5 Experimental Setup

In this section, we describe the settings of our experiments for privacy-aware text rewriting.

4.5.1 Datasets

In this paper we conduct experiments on three tasks, which can lead to potential social-good applications, namely obscuring gender, political slant and race of the authors.

Gender [Reddy and Knight, 2016] is a dataset of reviews from Yelp annotated with the gender of the authors, either male or female. The sentences with low indication of gender (likelihood of gender lower than 0.7) is filtered out.

Politics [Voigt et al., 2018] is a dataset of comments on Facebook posts from 412 members from the United States Senate and House. Each comment is associated

with the corresponding Congressperson's party affiliation as the sensitive attribute, $S \in \{\text{democratic}, \text{republican}\}$.

Race [Blodgett et al., 2016] is a dataset based on the dialectal tweets corpus (DIAL), including 59.2 million tweets. The tweets are categorized into African-American English (AAE) or Standard American English (SAE), which is highly correlated to the race of the author. The predictor takes into account both the content of the tweets and the geolocations of the the authors. We filter out the samples with predicted confidence lower than 80%, and tweets with less than 3 words. We consider race as sensitive information of the dataset. We also maintain the sentiment classification as a target task for this corpus to check if the sentiment information is still preserved after rewriting. The sentiment labels are derived from emojis which are associated with sentiments.

All the aforementioned corpora are split into four disjoint parts: **Class**, training corpus for sensitive attribute classifier; **Train**, training corpus for privacy-aware text rewriting; **Valid**, validation set; and **Test**, test set. The number of sentences for each split of these datasets are listed in Table 4.1.[1] The datasets cover some specific attributes for sensitive personal information. For other sensitive attributes, we recommend to collect datasets with corresponding attribute labels.

Dataset	Class	Train	Valid	Test
Gender [Reddy and Knight, 2016]	2.6M	200K	4K	4K
Politics [Voigt et al., 2018]	80K	200K	4K	4K
Race [Blodgett et al., 2016]	80K	100K	4K	4K

Table 4.1: Data splits of `Gender`, `Politics` and `Race`.

4.5.2 Models

We consider the following three models for privacy-aware text rewriting.

- **Back Trans** is the naive back translation model, considered as baseline.
- **Adv** is the model using adversarial training to obscure the sensitive attributes.
- **SMDSP** model use Sequential Maximal Deviation between Subgroup Predictions in training.

We also compare the quality of generated text of our systems with an open-domain **Paraphrase** generation system [Iyyer et al., 2018].

4.5.3 Implementation Details

We use Transformer [Vaswani et al., 2017] as the translation architecture in our experiments. We re-implement the transformer model based on OpenNMT [Klein et al.,

[1]The datasets are publicly available at https://github.com/xuqiongkai/PATR.

2017]. In our experiments, we use the same configurations, including 2 encoder and decoder layers, 256-dimensional word embedding and 256-dimensional hidden layers, drop out rate 0.1, label smoothing weight 0.1. All models use Beam Search decoding algorithm with beam size 5.

We train English-French machine translation (En-Fr) and French-English back-translation (Fr-En) using Europarl v7 from WMT15 [Bojar et al., 2015]. The words are tokenized using Moses tokenizer [Koehn et al., 2007]. Our translation system achieves the BLEU scores of 36.24% and 37.36% on En-Fr and Fr-En, respectively. The En-Fr model is used to generate the parallel corpus for all experiments.

4.5.4 Evaluation

The generated sentences are evaluated according to both linguistic quality of the sentences and obfuscation of the sensitive attribute. For each of these two aspects, we conduct automatic evaluation and human evaluation, respectively.

Linguistic Quality focuses on evaluating the quality of the results based on their semantic relevance to the original text and gramatical fluency of the generated sentences. We adopt four automatic evaluation metrics, BLEU, GLEU, METEOR and WMD. BLEU [Papineni et al., 2002] and GLEU [Wu et al., 2016] measure the n-gram matching between hypothesis and reference, where GLEU considers both precision and recall. METEOR [Banerjee and Lavie, 2005] further applies stemming and synonym matching. Word Mover Distance (WMD) [Kusner et al., 2015b] calculates the optimal transport distance between word embedding in original and generated sentences[2]. Intuitively, BLEU and GLEU evaluate fluency of the sentence as they are based on the quality of n-grams, while WMD measures semantic relevance as words can be regarded as atom semantic components of sentences.

We also conduct human evaluation to judge the fluency and relevance of the results. The criteria are as follows:
Grammar and Fluency [1-5].

- 5: Without any grammatical error;
- 4: Fluent and has one or two minor grammatical error that does not affect understanding;
- 3: Basically fluent and has three or more minor grammatical errors or one serious grammatical error that does not have strong impact on understanding;
- 2: Can not understand what is the meaning, but is still in the form of human language;
- 1: Not in the form of human language.

Coherence and Consistency [1-5].

[2]We use pre-trained word2vec model trained on Google News dataset from https://code.google.com/archive/p/word2vec/.

- 5: Accurate paraphrase with exact the same meaning of the source sentence;
- 4: Basically the same meaning of the source sentence but does not cover some minor content;
- 3: Cover part of the content of source sentence and has serious information loss;
- 2: Topic relevant but fail to cover most of the content of source sentence;
- 1: Topic irrelevant or even can not understand what it means.

Sensitive Attribute

- For `Gender` samples, please judge whether they are posted by male or female.
- For `Politics` samples, please judge whether they are from democratic or republican members from the United States Senate and House.
- For `Race` samples, please judge whether they are in African-American English or Standard-American English.

For each set of the results, two annotators are asked to judge the quality of the results between the scales of 1-5. The Kappa coefficients [McHugh, 2012] on `Gender`, `Politics` and `Race` are 0.45, 0.47 and 0.74, respectively.

Obfuscation evaluates the leakage of sensitive attributes of generated text. For automatic evaluation, we estimate the probability of sensitive attribute on generated sentences using a Logistic Regression with L2 regularization [Pedregosa et al., 2011]. For all the experiments, we use top 3K frequent words as features. Based on the prediction of classifier $p_i = P(S = i|X)$, we propose to evaluate the obfuscation of the results using the following three metrics:

1. **Entropy** evaluates the averaged entropy ($\sum_i -p_i \log p_i$) of all predictions. Higher Entropy indicates better less sensitive information leakage.
2. **P-Acc**, prediction accuracy, calculates the portion of correct prediction of the sensitive attribute. In the case of binary classification, the score is better if it is closer to 50%.
3. **M-Acc**, modification accuracy, calculates the label probabilities of source and generated sentences. If the probability of the sensitive attribute decreases after rewriting, the modification is accepted. M-Acc counts the rate of accepted sentence modifications.

In human evaluation, annotators are asked to judge the sensitive attribute values of 300 sampled sentences in test set. We use accuracy to evaluation the awareness of sensitive information by human and automatic annotators. Due to the fact that human judgments underperform automatic judgments (see Table 4.4), we rely more on automatic metric to evaluate the rewriting results.

4.6 Results and Analysis

We first conduct human evaluation and discuss their relation to automatic evaluation metrics with regard to semantic relevance, grammatical fluency and obfuscation. Then, we compare our privacy-aware models according to linguistic quality and obfuscation. Later on, we test the semantic loss of our models on the target task. Finally, we provide some sample outputs for case study.

4.6.1 Human Evaluation

Firstly, we ask human annotators to evaluate linguistic quality of Back Trans, Adv ($\alpha = 1$) and SMDSP ($\alpha = 1$), based on the rewriting results from 300 test samples, with regard to fluency (Flu) and relevance (Rel). We calculate the Pearson Correlation between human and automatic evaluation metrics. Table 4.2 shows the correlation of semantic relevance between human and automatic evaluation. WMD is clear winner among all automatic metrics across the three datasets. According to Table 4.3, GLEU is the measure that most correlated to human judgement in terms of fluency, though METEOR falls slight short on the gender corpus. Unsurprisingly, the widely used BLEU is the relatively less correlated to human perception, which was also observed in machine translation [Wu et al., 2016; Callison-Burch et al., 2006].

Exp	BLEU	GLEU	METEOR	WMD
Gender(Adv)	0.489	0.557	0.559	**0.651**
Gender(SMDSP)	0.414	0.507	0.511	**0.645**
Politics(Adv)	0.372	0.460	0.496	**0.573**
Politics(SMDSP)	0.358	0.474	0.476	**0.563**
Race(Adv)	0.311	**0.545**	0.532	0.127
Race(SMDSP)	0.242	0.386	0.367	**0.382**

Table 4.2: Correlation between semantic relevance and automatic evaluation metrics on `Gender`, `Politics` and `Race`. The most correlated automatic metrics are **bold**.

Exp	BLEU	GLEU	METEOR	WMD
Gender(Adv)	0.265	**0.287**	0.222	0.297
Gender(SMDSP)	0.192	0.231	0.186	**0.361**
Politics(Adv)	0.180	0.260	**0.277**	0.200
Politics(SMDSP)	0.149	**0.236**	**0.236**	0.231
Race(Adv)	0.168	0.403	**0.433**	0.333
Race(SMDSP)	0.068	**0.150**	0.124	0.046

Table 4.3: Correlation between fluency and automatic evaluation metrics on `Gender`, `Politics` and `Race`. The top two correlated automatic metrics are **bold and underlined** (the highest) and underlined (the second highest).

Secondly, we compare the performance of predicting sensitive information between human annotators and automatic classifiers. We ask human annotators to

	Gender	Politic	Race
Automatic	**77.3**	**93.7**	**82.7**
Human	66.0	60.3	71.0

Table 4.4: Comparison of human and automatic judgments on `Gender`, `Politics` and `Race`, with regard to their accuracy on the sensitive attribute.

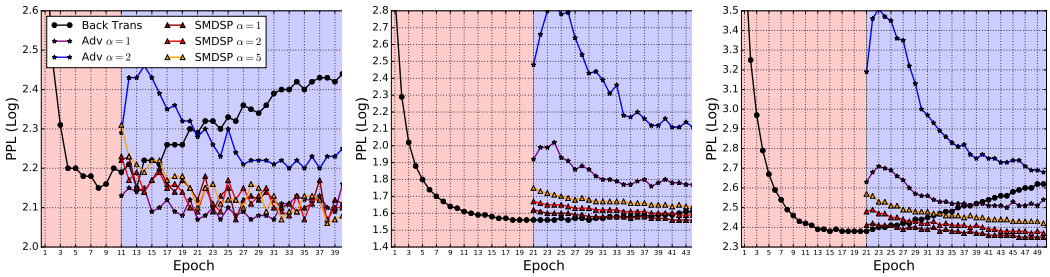
classify the sensitive attributes of 300 original sentences in test set. The accuracy of the annotations are illustrated in Table 4.4. To our surprise, human judgments are more than 10% worse than our classifiers on all the experiments. For `Politics`, we ask one more annotator for additional annotation and the accuracy of the annotation is still lower than 65%. After investigating the datasets, we found that a large proportion of samples are difficult for human annotators while our classifier can predict them correctly. For example, in `Gender`, human struggled in deciding whether "the food is delicious" and "the people were nice" are posted by male or female authors. For `Politics`, we observe several cases that human tends to annotate them with the opposite political slant when the sentences are in negative sentiment, while actually the speaker and the mentioned people support the same party, *e.g.*, "Patty Murray couldn't be any more dishonest than this!". Other examples like "today is such a wonderful day!" and "God bless you guys" are neutral to our annotators. Correctly annotating these samples might require extensive background in American politics. The top weighted words of male or female for `Gender`, democratic or republican for `Politics`, and SAE for `Race` are listed in Appendix A.2 to show the difficulty for human annotators to capture subtle indicators. To sum up, human annotators fail to incorporate subtle indicators into their decision, however, the classifiers manage to detect them.

The human evaluation studies conclude that *i)* we can rely on sensitive attribute classifiers for obfuscation evaluation, and *ii)* we should look at WMD for semantic relevance and GLEU for fluency.

4.6.2 Adversarial Learning vs. SMDSP

We conduct automatic evaluation on text generated by Back Trans, Adv and SMDSP. The overall observations are *i)* Back Trans provides a preliminary baseline for our task; *ii)* both Adv and SMDSP are able to reduce the leakage of sensitive information; and *iii)* SMDSP retains better linguistic quality, while Adv manages to preserve sensitive information.

We first compare the linguistic quality of the results in Table 4.5. The Back Trans outperforms both Adv and SMDSP on average because it does not cope with sensitive attributes in training. The performance of Adv model with the highest α obtains less than half GLEU than that of Back Trans. Although SMDSP with higher α also shows performance reduction, the quality of generated text are still competitive with Back Trans, with less than 10% score reduction. In particular, SMDSP with ($\alpha = 1$) achieves even higher GLEU on both `Politics` and `Race` than the baseline. We



(a) Gender (b) Politics (c) Race

Figure 4.1: Log perplexity(PPL) on valid set of `Gender`, `Politics` and `Race`. Red areas indicate pre-training epochs and Blue areas represent the epochs for privacy-aware training.

attribute this to the regularization effect of SMDSP on language modeling. Results of human evaluation are coherent to automatic evaluation, in Table 4.7. SMDSP achieves highest fluency results and competitive relevance results.

Then, we show the obfuscation performance in Table 4.6. Back Trans is a competitive baseline that obfuscates the classifiers to some extent. Adv and SMDSP are able to further reduce the obfuscation score on all three datasets. Generally, models with higher α achieve better obfuscation performance. Adv tend to be more aggressive on privacy preservation than SMDSP. However, we observe that Adv acquires better privacy preservation by sacrificing the linguistic quality, *e.g.*, Adv ($\alpha = 5$) basically chooses to 'keep silent' (produces almost no words) to protect the sensitive information on `Politics`[3]. We believe that generating totally non-sense sentences is too conservative for our task. On the other hand, SMDSP manages to protect sensitive attribute while keeping the semantic meaning as much as possible. For example, SMDSP ($\alpha = 1$) achieves both higher relevance score and better obfuscation score than Adv ($\alpha = 1$) on `Gender` and `Politics`.

Finally, we demonstrate the training stability of our models. The reconstruction losses of each model on validation set of `Gender`, `Politics` and `Race` are shown in Figure 4.1. We pre-train the back translation model for 10 epochs on `Gender` and 20 epochs on `Politics` and `Race`. Then, we train Adv model and SMDSP model based on the pre-trained model. We also include the pre-trained model with the same total number of training epochs in Black lines. After pre-training, Back-Trans models start to overfit and get slightly worse results on validation set. In most cases, the losses of Adv are higher than Transformer, and higher adversarial training intensity α decreases the performance of translation model. Adv ($\alpha = 5$) is not included in the plots, because their losses are out of the range. In contrast, SMDSP achieves better performance than Adv. The performance of SMDSP is even better than Back Trans on `Gender` and `Race`.

[3] All the generated sentences are empty on test set.

Dataset	Original	Back Trans	Adv	SMDSP
Gender	food is always delicious ! (Female)	the food is always yummy !	the food is always delicious !	the food is always amazing !
Gender	I went with my girlfriend , and another couple . (Male)	I went with my girlfriend , and another couple .	I went with my friend , and another couple .	I went with my friend , and another couple .
Politics	Sir Scott , a limited attack will make any solution for Syria nearly impossible . (Republican)	Sir Scott , a limited attack will make almost impossible to Syria .	Sir , a limited attack will almost impossible attack Syria .	Sir , a limited attack will make almost impossible for Syria .
Politics	love you U.S. senator al franken (Democratic)	love you U.S. senator al franken	help U.S. senator al franken	help U.S. senator al franken

Figure 4.2: Sample of original text, with sensitive attribute labels, and corresponding rewritten text using Back Trans, Adv ($\alpha = 1$) and SMDSP ($\alpha = 1$) on `Gender` and `Politics`.

4.6.3 Target Task Performance

We evaluate sentiment classification (Sent) as the target task and racial (Race) as sensitive attribute on the `Race`. The ideal rewriting should not change the predictive results of the non-sensitive attributes. Hence, we intend to observe better performance on Sent which implies better semantic preservation. As shown in Table 4.8, the prediction performance of both Race and Sent using Adv models decrease as the hyperparameter α increases. Such trend shows that Adv improves privacy preservation by obscuring the semantic meaning of the original text. In contrast, Risk models successfully decrease the accuracy on Race, while preserving the accuracy on Sent, showing the robustness of the model on preserving semantic meanings of the text.

4.6.4 Case Study

We demonstrate generated examples in Figure 4.2[4]. For `Gender`, Back Trans generates the words with clear tendency of gender, such as 'yummy' and 'girlfriend', while privacy-aware models use 'delicious', 'amazing' and 'friend' instead. For `Politics`, Adv and SMDSP skip the name after Sir to hide the political affiliation of the person. In the second example, Adv and SMDSP replace 'love you' with 'help' to reduce the political slant.

4.7 Summary

In this chapter, we have explored automatically obscuring personal attributes for privacy-aware text rewriting. Inspired by style transfer and obfuscation, we proposed to obscure the classifiable personal attributes in back-translation process and utilize

[4]Because the samples in `Race` are full of porny and violent words, they are excluded in the thesis.

i) adversarial training and *ii)* fairness-risk measurement to supervise the training of obscured back-translation models. Our results demonstrated the feasibility of obscuring some sensitive personal attributes with different strength, while we have observed that the rewrites with higher privacy protection scores are normally of lower linguistic and semantic quality.

Model	Gender			Politics			Race		
	GLEU	METEOR	WMD	GLEU	METEOR	WMD	GLEU	METEOR	WMD
Back Trans	**45.14**	**37.16**	**1.012**	37.29	**36.78**	**1.039**	23.09	26.94	1.460
Adv($\alpha = 1$)	44.11	36.76	1.023	29.44	33.55	1.125	12.94	18.07	**1.303**
Adv($\alpha = 2$)	40.29	34.34	1.117	23.20	26.82	1.261	12.75	18.39	1.430
Adv($\alpha = 5$)	22.98	23.32	1.561	N/A	N/A	N/A	9.67	17.03	2.242
SMDSP($\alpha = 1$)	44.17	36.69	1.031	**38.43**	36.59	1.044	**24.77**	**28.15**	1.483
SMDSP($\alpha = 2$)	43.10	35.84	1.062	38.01	36.36	1.056	23.95	27.49	1.501
SMDSP($\alpha = 10$)	41.54	35.09	1.101	36.40	35.96	1.069	23.10	26.99	1.531
SMDSP($\alpha = 100$)	40.90	34.64	1.122	36.84	35.64	1.082	22.74	26.81	2.242

Table 4.5: Automatic evaluation of linguistic quality on `Gender`, `Politics` and `Race`.

Model	Gender			Politics			Race		
	Entropy	P-Acc	M-Acc	Entropy	P-Acc	M-Acc	Entropy	P-Acc	M-Acc
Test(Ori)	0.5544	77.45	-	0.4873	93.05	-	0.3586	86.33	-
Back Trans	0.5617	72.45	48.90	0.5011	85.55	56.03	0.3960	74.68	62.35
Adv($\alpha = 1$)	0.5649	72.50	49.58	0.5026	84.90	57.25	0.4386	74.08	66.80
Adv($\alpha = 2$)	0.5644	70.23	52.73	0.5542	73.60	68.65	0.4623	73.40	69.13
Adv($\alpha = 5$)	0.5754	66.80	59.78	0.6931	50.00	93.15	0.5268	65.75	73.58
SMDSP($\alpha = 1$)	0.5711	71.80	50.18	0.5059	85.20	57.33	0.3989	74.85	62.48
SMDSP($\alpha = 2$)	0.5759	71.08	52.15	0.5066	84.95	58.35	0.4013	74.40	63.40
SMDSP($\alpha = 10$)	0.5768	70.88	53.05	0.5089	85.13	59.23	0.4007	74.08	63.65
SMDSP($\alpha = 100$)	0.5803	70.73	54.78	0.5129	85.08	59.90	0.4069	74.10	64.80

Table 4.6: Automatic evaluation of Obfuscation on `Gender`, `Politics` and `Race`.

Model	Gender		Politics		Race	
	Flu	Rel	Flu	Rel	Flu	Rel
Back Trans	4.68	4.09	**4.60**	**4.31**	4.31	3.88
Adv	4.66	4.13	4.42	4.01	3.84	3.53
SMDSP	**4.73**	**4.14**	**4.60**	4.21	**4.37**	**3.98**

Table 4.7: Human evaluation of fluency (Flu) and relevance (Rel) on `Gender`, `Politics` and `Race` based on the results of Back Trans, Adv ($\alpha = 1$) and SMDSP ($\alpha = 1$) with the scales of 1 to 5.

Model	Race	Sent
Test(Ori)	86.33	74.08
Back Trans	74.68	70.18
Adv($\alpha = 1$)	74.08	70.15
Adv($\alpha = 2$)	73.40	69.88
Adv($\alpha = 5$)	65.75	65.70
SMDSP($\alpha = 1$)	$74.85^{\dagger}$	69.88
SMDSP($\alpha = 2$)	74.40	$70.23^{\dagger}$
SMDSP($\alpha = 5$)	74.30	$70.15^{\dagger}$
SMDSP($\alpha = 10$)	74.08	70.60
SMDSP($\alpha = 100$)	74.10	$70.83^{\dagger}$

Table 4.8: Prediction accuracy (P-Acc) of classification results of race and sentiment classification task on `Race`. The results with higher accuracy than Back Trans are marked with daggers (†).

Chapter 5

Open-Domain Privacy-Aware Text Rewriting

In Chapter 4, we presented our research on obfuscating personal attributes for protecting privacy in text. The recent advance of natural language generation (NLG) system, such as the dialogue system of personal digital assistant, emerges urgent requirement of preventing the disclosure of diverse personal information. Moreover, as human, we have several natural and rational methods to mitigate the influence of privacy leakage in conversations, *e.g.*, deleting, obscuring, and steering the sensitive information. Inspired by these requirements, we discuss and amend two limitations of the setting in our work in Chapter 4, *i)* the representation of personal information is constrained to some pre-defined categories and *ii)* only obscuring is considered as rewriting strategy. In this Chapter, we further extend the privacy-aware text rewriting to more flexible and challenging setups. Firstly, we relax the personal information attributes to open-domain textual descriptions, which potentially provide infinite representations of private information. Secondly, inspired by [Strengers et al., 2020], we consider three rewriting strategies in this work: *Deleting*, *Obscuring*, and *Steering*. Overall, we collected a rewriting corpus with three strategies based on the dataset collected in Section 3.3. The rewriting model first detect the sensitive parts in the original utterance, and then rewrite them, as described in Section 5.3. We explore the feasibility of fine-tuning large-scale pre-trained language models for each rewriting strategy. Moreover, we observe that the strategies are correlated with semantic spaces in a knowledge graph, and therefore, propose a decoding method to incorporate such information into generated text, as described in Section 5.3.2.

5.1 Problem Statement

In this section, we discuss our extension of the settings for the privacy-aware text rewriting problem. We consider *i)* open-domain utterances to describe personal information and *ii)* multiple possible strategies for privacy-aware text rewriting.

Original:	Well, I do not like heights very much and I love animals.
Persona:	I am afraid of heights.
Deleting:	Well, I like animals.
Obscuring:	Well, I love animals but I feel uncomfortable at certain places.
Steering:	Well, I don't like depth and I like animals.

Table 5.1: An example of rewrites using *Deleting*, *Obscuring* and *Steering* as rewriting strategies, given the same original sentence (*Original*) and the corresponding personal information (*Persona*).

5.1.1 Open-Domain Personal Information Descriptions

The perception of sensitive personal information vary among the people with different culture backgrounds and usage scenarios. Users may be interested in protecting heterogeneous personal information, such as religion, relationship, living place, occupation and etc. [Li et al., 2015]. Considering the diverse information for description, we use complete sentences as personal information. Compared with labeled attributes, full sentences provide more flexibility to describe the diverse personal information and to satisfy different requirements of privacy protection for various users. As an example in Table 5.1, 'I am afraid of ___ .' could be served as one template to describe a series of fears of the people.

5.1.2 Multiple Rewriting Strategies

In order to eliminate the influence of disclosing personal information, there are multiple rewriting strategies could be conducted. Inspired by [Strengers et al., 2020], we consider three rewriting strategies in this work: *Deleting*, *Obscuring*, and *Steering*. The strategy *Deleting* simply removes all sensitive words from a text message. To make rewritten messages natural and easy-to-understand, *Obscuring* substitutes sensitive expressions for more abstract and general expressions. The proposed approaches in Chapter 4 are basically for *Obscuring*. In a similar manner to *Obscuring*, *Steering* replaces sensitive expressions by semantically relevant ones, which are *not* general or abstract expressions. Then, the challenge lies in the trade-off between readability and privacy protection of the rewrites. In our study in Section 5.2, we find that different rewriting patterns possess different levels of semantic relevance and privacy protection to the original inputs. As the rewrites shown in Table 5.1, *Deleting* eliminates more contents than the other two strategies, while *Obscuring* tends to generate the ones more semantically similar to the original sentence. *Steering* allows gentle modification on the semantic of the private information, while *Steering* tries to make such information less inferable. In real-world applications, it could be up to the users to select which strategy they prefer according to usage scenarios.

5.2 ODPAR Dataset

The first step toward privacy-aware text rewriting in open domain is to collect a corpus for the task. We constructed an **O**pen-**D**omain **P**rivacy-**A**ware **R**ewriting dataset, `ODPAR`, by collecting rewrites for the sentences that leak personal information (persona) in `PERSONA-LEAKAGE` [Xu et al., 2020], as described in Section 3.3. In that corpus, personal information is annotated with a privacy score. We collect the privacy-leaking sentence-persona pairs and filter out those with privacy score lower than 60% form `PERSONA-LEAKAGE`. For each sentence-persona pair, we ask annotators from Amazon Mechanical Turk[1] (AMTurk) to rewrite the original sentences using the following three rewriting strategies,

- ***Deleting***: removing the words or phrases that leak the provided personal information from the original sentences;
- ***Obscuring***: generalizing or blurring the words or phrases that leak the provided personal information from the original sentences;
- ***Steering***: replacing the words or phrases that leak the provided personal information with semantically similar expressions, which steer slightly away from the original meanings.

In our preliminary experiments, we observe that even though annotators endeavor to generate decent rewrites, many of them could not clearly identify and strictly stick to the required strategies. To make sure that the annotators are aware of the correct strategy, we wrap up 15 utterance-persona pairs as a batch and ask annotators to rewrite them using the required strategies. Then, we manually check the batches rewritten utterances, we only accept those that are written using the required strategy. The averaged acceptance rate of the rewrites is 47.97%, demonstrating the challenge of collecting a high-quality rewriting dataset with specific rewriting requirements. Finally, we split our dataset into train, valid and test sets with 600x3, 150x3 and 195x3 paralleled rewriting samples, respectively. We also split the dataset w.r.t. rewriting strategies into three subsets, namely **`DELETE`**, **`OBSCURE`** and **`STEER`**.

We analyze the each subset using averaged word length in sentences (Len.) and distinct unigrams divided by the total number of words (Dist.) [Li et al., 2016]. The statistics of the dataset is given in Table 5.2. `DELETE` tends to contain more concise rewrites, while `OBSCURE` and `STEER` includes slightly longer sentences than ORIGINAL. Although the average length increases, the Dist. score on `OBSCURE` and `STEER` are still ascending, compared with original sentences. This shows the high diversity of word/token usage in `OBSCURE` and `STEER`.

Then, we ask annotators to evaluate the quality of the rewrites, based on the grammatical fluency of the rewrites (Fluency), semantic relevance to the original sentence (Semantic), and privacy protection quality compared with corresponding personal information (Privacy) with scores scaled between 0 (the worst) and 3 (the

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

	Train		Valid		Test	
	Len.	Dist.	Len.	Dist.	Len.	Dist.
ORIGINAL	13.7	0.148	13.6	0.257	13.5	0.248
DELETE	8.0	0.190	8.4	0.298	8.5	0.279
OBSCURE	14.1	0.160	13.9	0.266	14.3	0.250
STEER	14.1	0.167	14.3	0.285	14.2	0.256

Table 5.2: Statistics of original sentence (ORIGINAL), rewrites with *Deleting*, *Obscuring* and *Steering* on train, valid and test set of corresponding subsets of the ODPAR, DELETE, OBSCURE, and STEER, using average length (Len.) and distinct token (Dist.)

best). Each sentence is evaluated by three annotators, hired from AMTurk. More details about the annotation guideline are provided in Appendix A.3. The overall fluency of human rewrites is very high. There is no free lunch for privacy-aware rewriting task, better privacy preservation is generally accompanied with relatively lower semantic scores.

	Fluency	Semantic	Privacy
DELETE	2.63	1.77	2.57
OBSCURE	2.59	2.35	1.58
STEER	2.59	0.67	2.67

Table 5.3: Human evaluation of grammatical fluency (Fluency), semantic relevance(Semantic) and privacy protection (Privacy) score of the rewrites with *Deleting*, *Obscuring* and *Steering* as rewriting patterns, scaled in [0-3].

5.3 Methodology

Privacy-aware rewriting is concerned with the protection of personal information. Therefore, a practical assumption is that descriptions of sensitive personal information are only visible to owners of such information, such as private textual descriptions in user profiles of social networks. In this work, we show that it is still feasible to achieve decent performance with the approaches requiring minimal training data.

As illustrated in Figure 5.1, we present a system for privacy-aware text rewriting, which is composed of a privacy-leakage detection module and a rewriting module. The detection module identifies sensitive words in a sentence by aligning them with those in a provided persona. The rewriting module is a sequence-to-sequence model that rephrases the given sentence according to one of the strategies (A): *Deleting*, *Obscuring*, and *Steering*.

Formally, given a text message $X = \langle x_1, ..., x_m \rangle$ with x_i from a vocabulary $\mathcal{V}$, a persona $P = \langle p_1, ..., p_n \rangle$ with $p_i \in \mathcal{V}$ describing sensitive personal information, the detection task aims to identify all words in X that refer to sensitive information

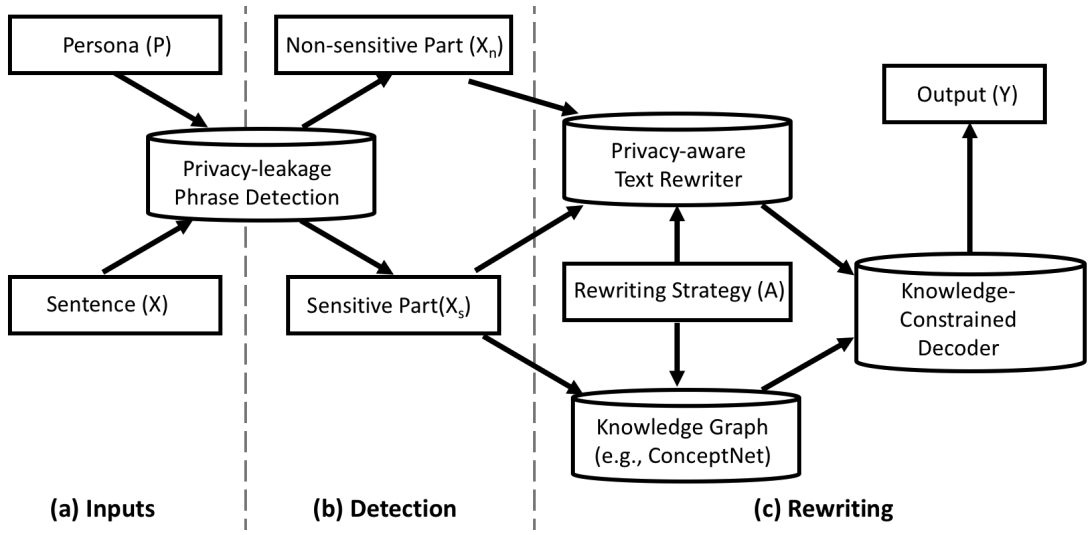


Figure 5.1: The workflow of our proposed knowledge guided privacy-aware rewriting system (KGPR). Given the original sentence (X) and persona (P) as inputs, the system *i)* detects the sensitive parts X_s and non-sensitive parts X_s of X; and *ii)* rewrites the text based on the required rewriting strategy (A) and X_s, X_n. Our proposed decoding model incorporates constraint knowledge as derived from A and X_s.

in P, and the rewriting task is to revise X into a message $Y = \langle y_1, ..., y_T \rangle$ by either *Deleting*, *Obscuring*, or *Steering*, such that *i)* the sensitive information in X is removed; *ii)* semantic relevance to non-sensitive part of X is maximally preserved; *iii)* Y is as natural as possible.

5.3.1 Privacy Detection by Alignment

We consider an open-domain privacy detection problem, given each message pairing with a persona describing the corresponding sensitive information. It is common in social networks and PAs that a software system knows which messages are sent by which interlocutor, as well as a list of sensitive personas of that interlocutor. We assume that there are pre-processing software components, such as the ones proposed in Section 3.4 [Xu et al., 2020], which are able to align privacy-leaking messages with the corresponding sensitive personas.

As we already know that sensitive information is described in a provided persona, we design a module to detect which part of the message conveys such information. We formulate it as an alignment problem between the words in X and the words in P. More specifically, we adopt linear sum assignment problem algorithm (LSAP) [Hessel et al., 2019b] for unsupervised image-sentence alignment to this word alignment problem.

Given a message X and a persona P, we construct a word-level association matrix $\mathbf{A} = \{a_{ij}\}_{n \times m}$, in which each entry a_{ij} is the similarity score between a word $x_i \in X$ and a word $p_j \in P$. The similarity score is calculated as the cosine similarity of

the contextual representation of the word pairs. Given the association matrix $\mathbf{A}$, an alignment is a set of edges in such a bipartite graph that maximizes the sum of similarity values, which is represented as a matching matrix $\mathbf{M}\{m_{ij}\}_{n\times m}$. We assume that each node has at most one edge because a word is unlikely associated with more than one words. Let a binary variable m_{ij} denote the alignment edge between a node i and a node j, the alignment problem is formulated as the following integer linear program (ILP).

$$\begin{aligned} sim(X,P) &= \max_{m_{ij}} \sum_{i,j} m_{ij} a_{ij} \\ s.t. \forall i, \sum_{j} m_{ij} \leq 1,\ \forall j, \sum_{i} m_{ij} &\leq 1 \end{aligned} \tag{5.1}$$

where the outcome of the ILP program can be viewed as the alignment score between X and P.

To compute similarity values between a pair of words, we map X and P to a sequence of vectors by using a neural encoder $g(\cdot)$, respectively. The encoder is implemented by applying a pre-trained BERT [Devlin et al., 2019b] model to map a word sequence to an embedding sequence, followed by using a linear layer to project each embedding into a shared embedding space between X and P. Given a projected embedding $\mathbf{e}_i$ of x_i and a projected embedding $\mathbf{e}_j$ of p_j, m_{ij} is calculated as the cosine similarity between $\mathbf{e}_i$ and $\mathbf{e}_j$, *i.e.*,

$$a_{ij} = \frac{\langle \mathbf{e}_i, \mathbf{e}_j \rangle}{\|\mathbf{e}_i\| \|\mathbf{e}_i\|}. \tag{5.2}$$

We train the encoder by optimizing a contrastive loss. The rationale behind this is that the alignment model should align persona better to original message, $sim(X_k, P_k)$, than to the rewritten text, $sim(Y_k, P_k)$, as original utterance should disclose more corresponding personal information. For each rewriting samples (X_k, Y_k, P_k) in dataset $\mathcal{D}$, the loss is calculated as

$$\mathcal{L}(X_k, Y_k, P_k) = \max\{0, \alpha - sim(X_k, P_k) + sim(Y_k, P_k)\} \tag{5.3}$$

where $sim(X_k, P_k)$ evaluates the similarity between original sentence and persona, and $sim(Y_k, P_k)$ estimates the similarity between rewritten sentence and persona, respectively. $\alpha \in \mathbb{R}^+$ is the margin of the loss. Then the loss on training set is the sum of all example losses $\mathcal{L}(\mathcal{D}) = \sum_{k=1}^{K} \mathcal{L}(X_k, Y_k, P_k)$. In this work, we use the subset `DELETE` for training detection model, instead of using the whole dataset, because the sensitive parts are more distinguishable by rewrites using deleting as the strategy.

5.3.2 Strategy-Specific Rewriting

Given identified sensitive words in messages, the rewriting task aims to rephrase the messages by following one of the three strategies. As the sensitive words are known, represented as a mask token, the rewriting module should learn how to rewrite

the sensitive part using the corresponding pattern. *Deleting* is relatively straightforward, as the generator could serve as a language model and paraphrased the masked sentence to a fluent one. Thus, we focus on discussing the methods for *Obscuring* and *Steering*.

Both *Obscuring* and *Steering* choose to remove sensitive information while maximally preserving original meanings. *Obscuring* considers more general and abstract expressions than sensitive expressions, while *Steering* favors expressions that are neither more general nor paraphrases of sensitive expressions. In another words, they define different semantic spaces of rewritten texts, w.r.t. given sensitive expressions. Therefore, the key challenge herein is how to map sensitive expressions into the target semantic space.

Large-scale transformer-based conditional language models trained on web-scale corpora are able to generate grammatically coherent and semantically relevant text [Lewis et al., 2020; Yan et al., 2020]. Such models are trained to compute the probability of an output sequence Y, given an input text X.

$$Pr(Y|X) = \prod_{t=1}^{T} Pr(y_t|X, y_{<t}) \tag{5.4}$$

In this work, we choose pre-trained BART [Lewis et al., 2020] to rewrite messages because its pre-training tasks learn to reconstruct input texts corrupted by arbitrary noising functions, which are close to our task setting. As there is only a handful of parallel data available for training, we expect minimal changes to the model such that we steer the generation process of the model only when it is likely to generate sensitive words.

In light of the above analysis, our key idea is to encourage substituting sensitive words for the ones from the target semantic space and penalize the use of sensitive words during decoding. For each sensitive word $x_i \in X_s$, we look up a set of substitution candidates S_{x_i} in ConceptNet [Speer et al., 2017] along the relations implied by the target rewriting strategy. For *Obscuring*, we consider *isA*, *CapableOf*, *HasPrerequisite*, *Desires*, and *SimilarTo*, while we select *Antonym*, *DistinctFrom*, *HasProperty* and *AtLocation* for *Steering*. The selection of target relations are based on the corpus analysis in Section 5.5.3. As we encourage selecting a word from S_{x_i} in place of x_i, the substitution candidate sets provide constraints to form a semantic space for the target strategy.

During decoding, we increase the probabilities of substitution candidates and decrease the probabilities of sensitive words. Let $\mathbf{v}$ denote the logit vector computed by the conditional language model at time t, we modify the logit vector by multiplying it with a masking vector for constraints $\mathbf{c}$.

$$Pr(y_t|X, y_{<t}) = \text{softmax}(\mathbf{v} \circ \mathbf{c}) \tag{5.5}$$

where $\circ$ denotes element-wise multiplication, *i.e.*, Hadamard product. Let X_s be the set of sensitive words from X, $\lambda_u \in [1, +\infty]$ and $\lambda_d \in [1, +\infty]$ denote a scaling up

and a scaling down hyper-parameter respectively, we have

$$c_x = \begin{cases} \lambda_u & x \in \bigcup_{x' \in X_s} S_{x'} - X_s \\ 1/\lambda_d & x \in X_s \\ 1 & \text{Otherwise} \end{cases}$$

In this way, the model intends to select substitution candidates if the corresponding logit computed by the LM is sufficiently large and the selection of sensitive words is disabled by a large λ_d. This method injects the hyper-parameters λ_u and λ_d only during beam search thus does not change model parameters.

5.4 Experimental Setup

In this section, we describe the baselines and evaluation metrics for detection module and rewriting module, respectively.

5.4.1 Privacy-Leakage Word Detection

Methods. We compare six methods in this study. *i)* Random guess (**RANDOM**) is a preliminary baseline by randomly estimating if a word is sensitive or not. *ii)* **TOKEN MATCH** removes those tokens from messages that also appear in the corresponding personas. *iii)* **BERT MATCH** applies the pre-trained BERT_{base} [Devlin et al., 2019b] to map messages and personas to word embedding sequences. For each pair of message and persona, it computes the cosine similarity of embeddings between each token from the message and each token from the persona. The score of a token in a message is its maximal similarity score w.r.t. tokens in the persona. It yields a ranking list of tokens in a message based on the scores. *iv)* **MEAN** computes a word similarity matrix in the same way as **BERT MATCH**. Instead of taking the maximum, it takes the average score for each token in a message. *v)* **OPT** applies optimal Transport [Hessel et al., 2019b] to optimize the soft alignments, where alignment weights are in $[0, 1]$. *vi)* **LSAP** applies the LSAP algorithm introduced in Section 5.3.1.

Evaluation Details. As the above methods produce ranking of tokens, we evaluate them by IR metrics: precision at K (**P@K**), R-Precision (**Rprec**), normalized discounted cumulative gain (**NDCG**) and mean average precision (**MAP**)[2]. We also tune a threshold θ for the best performing method and keep only the tokens above the threshold. Note that, stop words and punctuation are excluded in all those experiments.

[2]https://trec.nist.gov/trec_eval/

5.4.2 Privacy-Aware Rewriting

Methods. We consider the following methods as well as their variations. *i)* **COPY** merely copies input messages as system outputs. *ii)* **BackTrans** (Back Translation) is a widely used paraphrasing system that translates messages into a pivot language, *e.g.*, French, and then translate it back to English. The mixture models for diverse machine translation [Shen et al., 2019] with single expert is used as a baseline. *iii)* **SEQ2SEQ** [Sutskever et al., 2014][3] is pre-trained on PAWS-X [Yang et al., 2019] and fine-tuned on each rewriting pattern of our corpus.

In our work, we propose to use pre-trained language model BART [Lewis et al., 2020] and its variations, because in our preliminary experiments, BART without fine-tuning (**BART**) outperforms GPT2 [Radford et al., 2019] and T5 [Raffel et al., 2020] on our dataset. Its variations include *a)* fine-tune BART with messages as input on the train set (**Src**); *b)* fine-tune BART that takes non-sensitive parts of messages as input (**Mask**); *c)* fine-tune BART that takes additionally personas as input **Persona** [Wolf et al., 2019]; *d)* decoding with our semantic constraint-based method using ConceptNet (**Constraint**). We consider the variation *(d)* as our novel method.

Evaluation Details To evaluate the performance of our rewriting system, we utilize three metrics, SARI, Sim_s and Sim_p. SARI [Xu et al., 2016b] is widely used for paraphrasing and grammar error correction as it emphasizes differences between source and target sequences in terms of insert, keep, and delete operations. BERTScore [Zhang et al., 2019] was proved successful in measuring the semantic relevance of two sentences given their sentence representations by BERT. We apply BERTScore to measure the semantic relevance *i)* between hypothesis and input messages (sim_s); and *ii)* between hypothesis and persona as indication of privacy leakage (sim_p). Lower sim_p indicates better privacy protection, while higher sim_s means better semantic preservation. Although BLEU [Papineni et al., 2002] is widely used in machine translation, it does not reward removal of privacy-leaking tokens.

5.5 Experimental Results

In this section, we present our experimental results, aiming at demonstrating the best performing methods in our study can achieve descent performance in terms of privacy detection and rewriting with limited training data.

5.5.1 Privacy-leakage detection

Table 5.4 reports the ranking results of all methods in comparison. The large margin w.r.t. RANDOM indicates that all other methods are indeed effective in finding privacy-leaking tokens through comparison with personas. BERT-based methods outperform TOKEN MATCH with a wide margin. It is another evidence of demonstrating the strengths of pre-trained BERT in terms of measuring semantic relevance.

[3]We use Fairseq https://github.com/pytorch/fairseq as code base for SEQ2SEQ baseline.

Model	P@1	P@3	P@5	Rprec	NDCG	MAP
RANDOM	0.3333	0.3800	0.3560	0.3904	0.5775	0.4923
TOKEN MATCH	0.6533	0.4400	0.3120	0.4734	0.6507	0.5649
BERT MATCH	0.7200	0.5889	0.4227	0.6864	0.7424	0.7261
MEAN ($\alpha = 0.2$)	0.7200	0.5889	0.4267	0.7044	0.7427	0.7279
OPT ($\alpha = 0.2$)	0.7200	**0.5911**	0.4267	0.7122	0.7440	0.7313
LSAP ($\alpha = 0.4$)	**0.7333**	**0.5911**	**0.4280**	**0.7156**	**0.7487**	**0.7373**

Table 5.4: Experimental results of privacy leakage token detection using random guess (RANDOM), exact token match (TOKEN MATCH), BERT MATCH, and alignment models (MEAN, OPT and LSAP).

Furthermore, optimizing alignments between tokens lead to further improvement, as demonstrated by OPT and LSAP. LSAP is slightly better because it removes noise by assuming each token is linked to at most one token in personas.

In Table 5.5, we illustrate the threshold selection process for our best detection model, *i.e.*, LSAP ($\alpha = 0.4$). With the increase of threshold θ, precision increases and recall decreases. Although the F1 score of the system with lower threshold θ is higher, we find the detection system tend to mask out useful information in the original sentence, which harms the performance in rewriting. In our system, we target on a more reliable detection system and set $\theta = 0.7$. Under such setting, LSAP reaches an F-1 score of 68.57% while having a relatively high precision of 92.31% on the valid set.

	Dev			Test		
θ	P (%)	R(%)	F-1(%)	P(%)	R(%)	F-1(%)
0.40	64.62	88.79	74.80	55.60	88.94	68.46
0.50	72.73	82.12	77.14	64.49	82.06	72.22
0.60	81.12	70.00	75.15	72.56	69.29	70.89
0.70‡	92.31	54.55	68.57	77.31	49.14	60.09
0.80	99.05	31.52	47.82	87.61	24.32	38.08

Table 5.5: The comparison of privacy detection model using various thresholds $\theta \in [0.4, 0.8]$. We use precision(P), recall(R) and F-1 score for the detected sensitive tokens.

5.5.2 Privacy-Aware Rewriting

As shown in Table 5.6, SARI favors methods achieving a good trade-off between semantic relevance to input messages and privacy leakage. Therefore, we prefer SARI as the main metric for automatic evaluation. The overall performance of the fine-tuned models with a specific rewriting strategy is marginally better than para-

phrasing baselines, especially for the models trained on OBSCURE and STEER. This shows that, even with limited amount of training samples, fine-tuning the existing pre-trained generation model is effective on our task. Introducing persona as input to the system exaggerates the leakage of persona. We will leave the discussion of effective privacy-preserving neural architecture in our future work. Knowledge constraints further improves the performance for the models use original source sentences, however, it has limited contributions to the models use masked messages as input. We attribute this to the fact that Mask has already eliminated the semantic of sensitive parts and it is hard for a generator to recover the obscured or steered paraphrase expressions without clues to the original semantics.

	DELETE			OBSCURE			STEER		
Model	SARI	Sim_s	Sim_p	SARI	Sim_s	Sim_p	SARI	Sim_s	Sim_p
COPY	17.89	0.999	0.579	23.22	0.999	0.579	23.99	0.999	0.579
SEQ2SEQ	28.00	0.492	0.438	24.33	0.544	0.476	27.01	0.582	0.481
BackTrans	30.76	0.739	0.573	25.89	0.739	0.573	27.56	0.739	0.573
BART	19.01	0.931	0.574	23.61	0.931	0.574	24.99	0.931	0.574
Src	**31.03**	0.927	0.581	35.68	0.885	0.576	43.88	0.899	0.583
Src+Persona	-	-	-	35.35	0.819	0.710	40.61	0.827	0.735
Src+Constraint	-	-	-	**36.98**	0.870	0.565	**46.55**	0.890	0.580
Mask	**39.55**	0.807	0.491	40.57	0.790	0.506	**44.34**	0.792	0.504
Mask+Persona	-	-	-	37.55	0.774	0.695	40.66	0.778	0.711
Mask+Constraint	-	-	-	**40.69**	0.791	0.506	44.22	0.792	0.503

Table 5.6: The comparison of rewriting models on DELETE, OBSCURE and STEER, using SARI, semantic similarity (Sim_s) and persona similarity (Sim_p).

5.5.3 Knowledge Constraint Analysis

We analyze correlations between rewriting strategies and knowledge constraints on the training set. We collect the direct neighbors of the detected sensitive tokens in ConceptNet 5.5 [Speer et al., 2017]. We count the numbers of the tokens in the rewriting references that also co-occur in the neighbors of privacy-leaking tokens[4]. As there are more than 30 possible relations in ConceptNet, we give detailed report in Table 5.7. The key finding is that some relations are more relevant to *Obscuring*, such as *IsA* and *CapableOf*, while some other relations are more relevant to *Steering*, such as *Antonym* and *DistinctFrom*. We consider the relations *i)* with a total of more than 10 hits and *ii)* have a clear preference on a rewriting pattern (at least two times the number of the second highest pattern). As a result, we select five relations, {*IsA*, *CapableO*, *HasPrerequisite*, *Desires* and *SimilarTo*}, as constraints for *Obscuring*, and four relations, {*Antonym*, *DistinctFrom*, *HasProperty*, and *AtLocation*}, as constraints for *Steering*. The total hits of the selected knowledge constraint sets on rewriting

[4]We conduct the statistics only on train set to avoid setting biases on validation and test sets.

patterns are reported in Table 5.8. Our strategic semantic space is strongly correlated to corresponding rewrites by human.

In Table 5.9, we conduct knowledge constraint analysis on the results of various system outputs, to explain and demonstrate the success of the fine-tune models and the proposed constrained decoding method. BART and Src baselines struggles to incorporated semantic in obscuring or steering space. The models (Src) trained on corresponding subspace manage to capture the semantic for a specific strategy. Incorporating the constraints to decoders almost doubles the hit counts.

5.5.4 Human Evaluation

We evaluate the quality of system outputs using, grammatical fluency (Fluency), semantic relevance (Semantic) with regard to non-sensitive part of the original sentence, and privacy protection level (Privacy), scaled from 0 (the worst) to 3 (the best). Additionally, we use correct pattern rate (CPR), which calculates the rate of the rewrites using the correct rewriting pattern. We randomly picked 80 samples from the test set for human evaluation. More details on annotation guideline are given in Appendix A.3.

The results are reported in Table 5.10. Overall, M_d using masked inputs (Mask) protects private information better than M_d using original source inputs (Src), as sensitive information is masked out from input, while the later one achieves the best fluency and semantic scores, as the model uses the most information in original sentences. The models M_s and M_o fine-tuned on `STEER` and `OBSCURE` manage to protect privacy with acceptable semantic loss, and M_s works better on average with regard to privacy protection score and correct pattern rate. Knowledge constraints (KC) improves the privacy protection scores on all four settings. In particular, KC significantly improves the CPR on OBS experiments. Surprisingly, semantic and privacy scores improves at the same time when adding KC to the M_s, Src vs. Src+KC and Mask vs. Mask+KC. This result further demonstrates the effectiveness of the KC decoding method.

5.6 Summary

In this chapter, we have proposed a task of privacy-aware text rewriting on open-domain personal information descriptions. A new dataset `ODPAR` with three rewriting strategies were collected for the task. We conducted intensive experiments to show the effectiveness of detect-and-rewrite framework on the new task. We have also proposed a strategy-specific decoding method integrating knowledge graph as constraints to better incorporate the corresponding strategies. The work in this chapter demonstrated the feasibility of protecting privacy in text via rewriting, given open-domain textual descriptions of personal information as control signals.

Relation Edge	DELETE	OBSCURE	STEER	Selected
RelatedTo	14	147	103	-
IsA	1	100	31	*Obscuring*
CapableOf	4	47	23	*Obscuring*
Synonym	1	29	34	-
UsedFor	0	32	20	-
Antonym	1	11	41	*Steering*
DistinctFrom	0	8	26	*Steering*
HasProperty	1	4	16	*Steering*
AtLocation	2	4	15	*Steering*
FormOf	5	9	3	-
HasPrerequisite	1	10	5	*Obscuring*
Desires	2	11	3	*Obscuring*
SimilarTo	0	11	3	*Obscuring*
HasA	0	7	4	-
MotivatedByGoal	2	3	2	-
ReceivesAction	0	6	2	-
DerivedFrom	0	6	0	-
MannerOf	0	4	1	-
NotDesires	0	4	1	-
HasFirstSubevent	0	1	2	-
HasLastSubevent	0	2	1	-
InstanceOf	0	3	0	-
CausesDesire	0	2	0	-
PartOf	2	0	0	-
DefinedAs	1	1	1	-
HasSubevent	1	1	0	-
Causes	1	1	1	-
HasContext	0	1	1	-
EtymologicallyDerivedFrom	0	1	0	-
NotCapableOf	0	1	0	-
genre	0	1	0	-
EtymologicallyRelatedTo	0	1	0	-

Table 5.7: Counts of words appear in both knowledge constraints and rewrites in three rewriting subsets, DELETE, OBSCURE and STEER. The selected relation types for the corresponding rewriting strategies are noted as *Obscuring* and *Steering*.

Constraint	Set	DELETE	OBSCURE	STEER
Obscure	train	6	**152**	58
Obscure	dev	1	**35**	6
Steer	train	4	17	**66**
Steer	dev	0	6	**8**

Table 5.8: The number of words that appear in both knowledge constraint sets and rewrites in corresponding datasets, DELETE, OBSCURE and STEER.

Model	Dataset	Constraint Set	
		Obscure	Steer
BART	Other	3	7
M_d (Src)	DELETE	0	0
M_o (Src)	OBSCURE	**40**	4
M_o (Src+Constraint)	OBSCURE	**73***	8
M_s (Src)	STEER	20	**23**
M_s (Src+Constraint)	STEER	28	**45***

Table 5.9: The number of words that appear in both knowledge constraint sets, Obscure and Steer and outputs of models trained on datasets, DELETE (M_d), OBSCURE (M_o), and STEER (M_s). BART is pre-trained on other large-scale out of domain (OOD) datasets. We also compare the model with decoding enhanced by corresponding constraint strategy (Src + Constraint).

Model	Test Set	Flu.	Sem.	Pri.	CPR(%)
M_d (Src)	DELETE	2.897	2.718	0.590	37.23
M_d (Mask)	DELETE	2.407	2.272	1.889	53.08
M_o (Src)	OBSCURE	2.614	2.506	1.016	66.33
M_o (Src+KC)	OBSCURE	2.667	2.497	1.099	66.38
M_o (Mask)	OBSCURE	2.321	2.222	1.914	80.45
M_o (Mask+KC)	OBSCURE	2.420	2.173	1.977	82.74
M_s (Src)	STEER	2.341	2.183	1.524	74.37
M_s (Mask)	STEER	2.439	2.246	2.118	73.90
M_s (Src+KC)	STEER	2.500	2.493	1.832	76.83
M_s (Mask+KC)	STEER	2.346	2.299	2.185	76.45

Table 5.10: The comparison of rewriting models fine-tuned on DELETE (M_d) OBSCURE (M_o) and STEER (M_s), using grammatical fluency(Flu.), semantic relevance(Sem.) and privacy protection (Pri.) score and correct pattern rate (CPR).

Chapter 6

Conclusion

In this chapter, we summarize the contributions of this thesis for privacy protection in conversations. We also present some potential research directions for future work.

6.1 Summary

This thesis aimed to solve the challenge of protecting potential privacy leakage in conversations. In particular, we proposed the new problem and discussed its urgent requirement. Then, we proposed detect-and-rewrite framework to solve this problem. Empirically, we collected datasets and developed models for detection and rewriting tasks, which demonstrates the feasibility of automatic privacy protection for conversations. We summarize our contributions as:

- **Proposing a new privacy protection challenge.** In Chapter 1, we introduced an emerging challenge for protecting privacy in conversations. We unveiled the requirement to protect privacy in conversations for social media applications and digital personal assistant. To solve the new challenge, we proposed a detect-and-rewrite framework for the new task.
- **Privacy leakage detection in conversations.** In Chapter 3, we discussed how to detect the utterances with potential privacy leakage risk. We proposed a weakly supervised learning framework to learn inference models. In particular, we collected a new dataset for detecting privacy leakage in conversations. We also developed new models that outperforms existing alignment models. Moreover, we observed that more advanced dialogue system tends to incorporate more personal information, leading to higher risk of privacy concerns, and our models managed to detect most of them. Finally, we provided a working demo that detected the privacy leakage and reported the risky cases to the authorized users.
- **Obscuring personal attributes via rewriting.** In Chapter 4, we explored how to obscure the sensitive personal attributes in texts by rewriting. We utilized back-translation framework and eliminated the sensitive information in the back translate step. Adversarial training and fairness-risk measurement were explored to train the rewriting models towards more neutralized outputs.

- **Open-domain privacy-aware rewriting.** In chapter 5, we extended the rewriting setting to open-domain and discussed three plausible rewriting strategies. In particular, we first collected a new rewriting datasets with three rewrites given the original utterance and disclosed personal information. Then, we fine-tuned large-scale pre-trained language models on rewriting strategies. Moreover, we observed the rewriting strategies are correlated to a semantic constraint space in a knowledge graph and incorporated the constraints to the decoding algorithm. The proposed models achieved decent rewriting results with different focuses. Users could select the preferable model with regard to semantic preservation, privacy protection and grammatical fluency.

6.2 Discussion and Future Work

While this thesis provides a comprehensive solution to privacy protection in conversations, the security and ethical issues of machine learning models, such as language generation models, start to attract more and more attentions from both research community and our society. We consider the following future work which may further benefit the related researches:

- **More generalized privacy protection.** This thesis assumes that the personal information that requires preservation is given. However, in some cases, the security service providers may not have the access to users' sensitive personal information. Herein, we may explore a more general system, which can detect and rewrite the sensitive information based on commonsense knowledge or meta-settings encoded in the models.

- **More personalized privacy protection.** Users with diverse backgrounds may have different requirements for privacy protection. Even for the information with exactly the same description, some people may consider them sensitive while others may not. Therefore, a personalized privacy protection system is necessary. This problem may be solved by using cold-start recommendation systems Schein et al. [2002] that generate or retrieve the sensitive personal information based on some initial questions or setups given by the users.

- **Integrating rewriting modules to real-world applications.** In Section 3.7, we demonstrated a real-world application for privacy leakage detection. This application requires the authorized users not automatic models to modify the utterances. In current stage, human users may take the most responsibility to protect their privacy in conversations, as human is still more reliable than machine learning models with regard to generation tasks. In future, with more conversations conducted by automated machine or digital personal assistant, there will be emergent requirements for privacy-aware text rewriting for those automatically generated utterances. Furthermore, we may incorporate the privacy protection module to in existing language generation systems.

Appendix

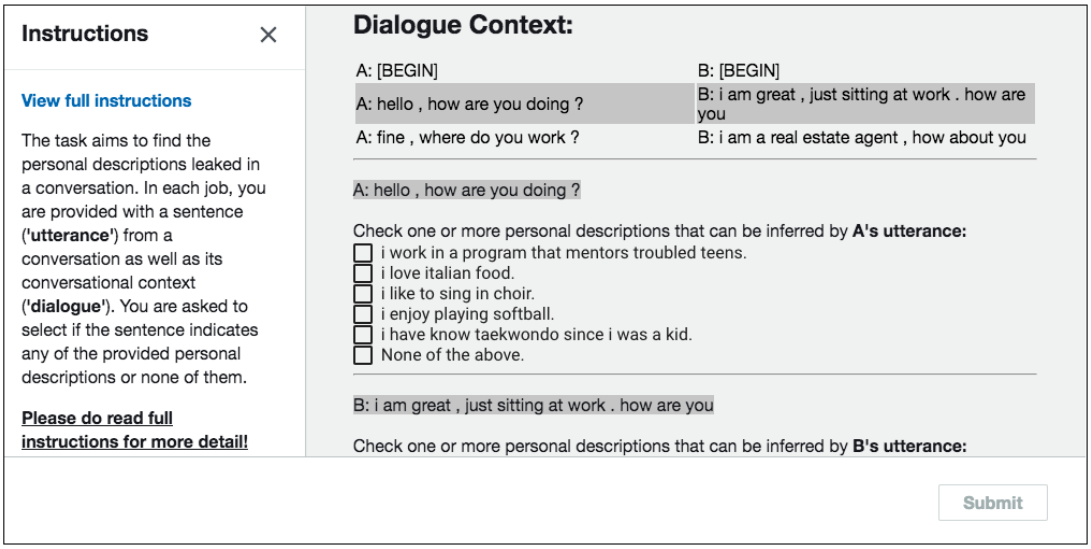


Figure A.1: Task screenshot for utterance-persona alignment annotation.

A.1 Details for PERSONA-LEAKAGE Collection

Starting from test set of PERSONA, our dataset basically tops up two annotations on test sets, alignment annotations on utterance-persona pairs and sensitivity annotations on all personal information statements. For both parts, we use Amazon Mechanical Turk (MTurk)[1] for crowdsourcing. We only accept results from the qualified annotators that *i)* have more than 90% HIT acceptance rate, *ii)* have finished more than 100 HITs, *iii)* locate in America. For further quality control, we reject 2.1% and 2.0% unreliable HITs for alignment annotation and sensitivity annotation respectively by automatically rejecting HITs that are *i)* not completed or *ii)* inconsistent in answers.

[1] https://requester.mturk.com/

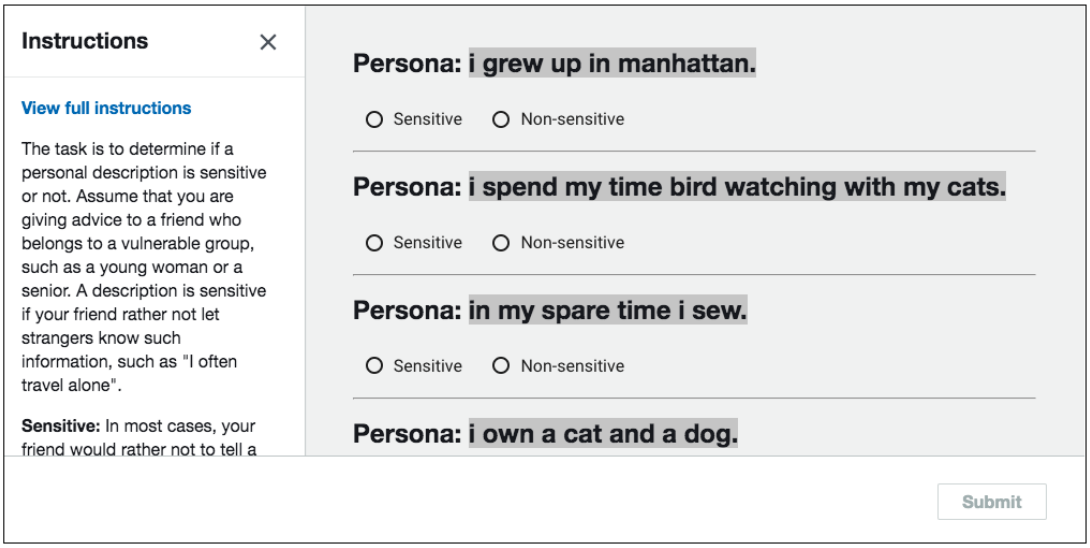


Figure A.2: Task screenshot for personal information sensitivity annotation.

For alignment annotations, annotators were instructed to *"find the personal descriptions leaked in a conversation"* by *"select if the sentence indicates any of the provided personal descriptions or none of them"*, see task screenshot in Figure A.1.

For sensitivity annotations, annotators were instructed to *"give advice to a friend who belongs to a vulnerable group"*, see task screenshot in Figure A.2. Sensitive information is defined as the one that *"your friend rather not let strangers know"*.

- **Sensitive:** In most cases, your friend would rather not to tell a stranger such information. Otherwise it will do more harm than good if the information is utilized by malicious people.
- **Non-sensitive:** In most cases, it is safe for your friend to share such information with strangers.

A.2 Sensitive Attribute Classifier

We list the top 20 weighted words for each sensitive attribute classifier. Top weighted features for AAE are full of bully and sexism words, which are not appropriate to be demonstrated in the thesis.

A.3 Linguistic Quality Annotation

In this section, we demonstrate the questions that test the linguistic quality of the rewrites. The questions and annotation guidelines are listed follow,
Q1: How is the grammaticality and fluency of the rewritten sentence?

- 3: No grammatical error.

Model	Top Weighted Words
Gender (Female)	husband, boyfriend, yummy, cute, hubby, lovely, BF, fabulous, gorgeous, delish, beautiful, love, salon, loved, massage, gross, spa, adorable, we, soooo
Gender (Male)	wife, girlfriend, buddy, gf, notch, solid, value, beers, excellent, outstanding, steaks, desert, ribeye, dude, brisket, beer, average, bucks, damn, guys
Politic (Democratic)	thank, Bernie, Warren, amy, Elizabeth, trump, democratic, al, Hillary, Booker, women, Sanders, patty, violence, drugs, schumer, debbie, Minnesota, Cory, democrats
Politic (Republican)	Obama, McCain, rand, mia, Paul, conservative, obamacare, rubio, sir, praying, constitution, god, gowdy, Marco, trey, tax, republican, tom, spending, Devos
Race (SAE)	're, haha, guys, seriously, hahaha, perfect, excited, 30, such, makes, Haha, does, someone, are, sucks, awesome, literally, snapchat, actually, everyone

Table A.1: Top weighted words of sensitive attribute classifiers.

- 2: Minor grammatical errors that do not affect understanding.
- 1: Hard to derive the meaning but still a human language in English.
- 0: Empty sentence or not English.

Q2: How is the semantic relevance of the rewritten sentence to the non-sensitive part of the original sentence?

- 3: Accurately preserves the meaning of the original sentence.
- 2: Basically the same meaning but does not cover some minor content.
- 1: Has a minor resemblance to the meaning of the original sentence, however, it is also misleading.
- 0: Empty sentence or does not reflect the meaning of the original sentence at all.

Q3: How is the privacy protection degree of the rewritten sentence?

- 3: The provided personal information cannot be inferred by the rewritten sentence at all.
- 2: Hard to derive the personal information, but rewrite is weakly associated to the personal information.
- 1: Part of the personal information can be inferred from the rewritten sentence.
- 0: The personal information is fully contained in the rewritten sentence, directly or through paraphrasing.

Q4: Do you recognize the rewrite follows the correct pattern? Options: [Yes] or [No].

Bibliography

What is privacy? Australian Government. https://www.oaic.gov.au/privacy/your-privacy-rights/what-is-privacy,[Lastaccesed:01/06/09]. (cited on page 7)

Agarwal, A.; Beygelzimer, A.; Dudík, M.; Langford, J.; and Wallach, H., 2018. A reductions approach to fair classification. *arXiv preprint arXiv:1803.02453*, (2018). (cited on page 37)

Banerjee, S. and Lavie, A., 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the ACL workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 65–72. (cited on page 40)

Bannard, C. and Callison-Burch, C., 2005. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, 597–604. (cited on page 14)

Bender, E. M.; Gebru, T.; McMillan-Major, A.; and Shmitchell, S., 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 610–623. (cited on page 3)

Berardi, G.; Esuli, A.; Macdonald, C.; Ounis, I.; and Sebastiani, F., 2015. Semi-automated text classification for sensitivity identification. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 1711–1714. (cited on page 13)

Bitaab, M.; Cho, H.; Oest, A.; Zhang, P.; Sun, Z.; Pourmohamad, R.; Kim, D.; Bao, T.; Wang, R.; Shoshitaishvili, Y.; et al., 2020. Scam pandemic: How attackers exploit public fear through phishing. In *2020 APWG Symposium on Electronic Crime Research (eCrime)*, 1–10. IEEE. (cited on page 2)

Blodgett, S. L.; Green, L.; and O'Connor, B., 2016. Demographic dialectal variation in social media: A case study of african-american english. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1119–1130. (cited on page 39)

Bo, H.; Ding, S. H. H.; Fung, B. C. M.; and Iqbal, F., 2021. ER-AE: Differentially private text generation for authorship anonymization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 3997–4007. (cited on page 11)

Bojar, O.; Chatterjee, R.; Federmann, C.; Haddow, B.; Huck, M.; Hokamp, C.; Koehn, P.; Logacheva, V.; Monz, C.; Negri, M.; Post, M.; Scarton, C.; Specia, L.; and Turchi, M., 2015. Findings of the 2015 workshop on statistical machine translation. In *Proceedings of the 2015 Workshop on Statistical Machine Translation*. (cited on page 40)

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al., 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, (2020). (cited on page 15)

Callison-Burch, C.; Osborne, M.; and Koehn, P., 2006. Re-evaluation the role of bleu in machine translation research. In *11th Conference of the European Chapter of the Association for Computational Linguistics*. (cited on page 42)

Calmon, F.; Wei, D.; Vinzamuri, B.; Ramamurthy, K. N.; and Varshney, K. R., 2017. Optimized pre-processing for discrimination prevention. In *Proceedings of Advances in Neural Information Processing Systems*, 3992–4001. (cited on page 37)

Chawdhry, A. A.; Paullet, K.; and Douglas, D. M., 2013. Data privacy: Are we accidentally sharing too much information? In *Proceedings of the Conference for Information Systems Applied Research ISSN*, vol. 2167, 1508. (cited on page 1)

Chiu, J. P. and Nichols, E., 2016. Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, 4 (2016), 357–370. (cited on page 15)

Chung, J.; Gulcehre, C.; Cho, K.; and Bengio, Y., 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning, December 2014*. (cited on page 15)

Competition, A.; Commission, C.; et al., 2020. Targeting scams: report of the accc on scam activity 2020. (2020). (cited on page 2)

Dai, Z. and Callan, J., 2019. Deeper text understanding for ir with contextual neural language modeling. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 985–988. (cited on page 25)

Dathathri, S.; Madotto, A.; Lan, J.; Hung, J.; Frank, E.; Molino, P.; Yosinski, J.; and Liu, R., 2019. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*. (cited on page 18)

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K., 2019a. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. (cited on pages 9, 15, 16, 23, and 30)

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K., 2019b. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. (cited on pages 54 and 56)

Dinan, E.; Logacheva, V.; Malykh, V.; Miller, A.; Shuster, K.; Urbanek, J.; Kiela, D.; Szlam, A.; Serban, I.; Lowe, R.; et al., 2019. The second conversational intelligence challenge (convai2). *arXiv preprint arXiv:1902.00098*, (2019). (cited on pages 28 and 29)

Dwork, C.; Hardt, M.; Pitassi, T.; Reingold, O.; and Zemel, R., 2012. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, 214–226. (cited on page 9)

Elazar, Y. and Goldberg, Y., 2018. Adversarial removal of demographic attributes from text data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 11–21. (cited on page 17)

Emmery, C.; Manjavacas, E.; and Chrupała, G., 2018. Style obfuscation by invariance. In *Proceedings of the 27th International Conference on Computational Linguistics*, 984–996. (cited on page 35)

Fernandes, N.; Dras, M.; and McIver, A., 2019. Generalised differential privacy for text document processing. In *International Conference on Principles of Security and Trust*, 123–148. Springer. (cited on page 8)

Fu, Z.; Tan, X.; Peng, N.; Zhao, D.; and Yan, R., 2018. Style transfer in text: Exploration and evaluation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32. (cited on page 15)

Ganin, Y. and Lempitsky, V., 2015. Unsupervised domain adaptation by backpropagation. In *Proceedings of International Conference on Machine Learning*, 1180–1189. (cited on page 37)

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y., 2014. Generative adversarial nets. In *Advances in neural information processing systems*, 2672–2680. (cited on page 36)

Hardt, M.; Price, E.; and Srebro, N., 2016. Equality of opportunity in supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 3323–3331. (cited on page 9)

Hessel, J.; Lee, L.; and Mimno, D., 2019a. Unsupervised discovery of multimodal links in multi-image, multi-sentence documents. In *Proceedings of the 2019 EMNLP-IJCNLP*, 2034–2045. (cited on pages 12 and 26)

Hessel, J.; Lee, L.; and Mimno, D., 2019b. Unsupervised discovery of multimodal links in multi-image, multi-sentence documents. 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)*, 2034–2045. (cited on pages 53 and 56)

Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. *Neural computation*, 9 8 (1997), 1735–1780. (cited on page 15)

Holstein, K.; Wortman Vaughan, J.; Daumé III, H.; Dudík, M.; and Wallach, H., 2018. Opportunities for machine learning research to support fairness in industry practice. In *The Workshop on Critiquing and Correcting Trends in Machine Learning conducted at the 32nd Conference on Neural Information Processing Systems, Montreal, Canada*. (cited on page 7)

Iyyer, M.; Wieting, J.; Gimpel, K.; and Zettlemoyer, L., 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1875–1885. (cited on page 39)

Kessler, G. C., 2010. An overview of cryptography. *Online: http://www.garykessler.net/library/crypto.html,[Last accesed: 01/10/09]*, (2010). (cited on page 7)

Klein, G.; Kim, Y.; Deng, Y.; Senellart, J.; and Rush, A. M., 2017. Opennmt: Open-source toolkit for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. (cited on pages 38 and 39)

Koehn, P.; Hoang, H.; Birch, A.; Callison-Burch, C.; Federico, M.; Bertoldi, N.; Cowan, B.; Shen, W.; Moran, C.; Zens, R.; et al., 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Demo and Poster sessions*, 177–180. (cited on page 40)

Koehn, P. et al., 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, vol. 5, 79–86. Citeseer. (cited on page 16)

Kusner, M.; Loftus, J.; Russell, C.; and Silva, R., 2017. Counterfactual fairness. *Advances in Neural Information Processing Systems 30 (NIPS 2017) pre-proceedings*, 30 (2017). (cited on page 9)

Kusner, M.; Sun, Y.; Kolkin, N.; and Weinberger, K., 2015a. From word embeddings to document distances. In *Proceedings of the 33rd ICML*, 957–966. (cited on pages 8, 12, and 26)

Kusner, M.; Sun, Y.; Kolkin, N.; and Weinberger, K., 2015b. From word embeddings to document distances. In *International Conference on Machine Learning*, 957–966. (cited on page 40)

Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; and Soricut, R., 2019. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*. (cited on page 30)

Lee, K.-H.; Chen, X.; Hua, G.; Hu, H.; and He, X., 2018. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision*, 201–216. (cited on pages 12 and 25)

Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L., 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7871–7880. (cited on pages xvii, 15, 16, 55, and 57)

Li, J.; Galley, M.; Brockett, C.; Gao, J.; and Dolan, W. B., 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 110–119. (cited on page 51)

Li, J.; Jia, R.; He, H.; and Liang, P., 2018a. Delete, retrieve, generate: a simple approach to sentiment and style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1865–1874. (cited on page 17)

Li, Y.; Baldwin, T.; and Cohn, T., 2018b. Towards robust and privacy-preserving text representations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 25–30. (cited on pages 10 and 17)

Li, Y.; Li, Y.; Yan, Q.; and Deng, R. H., 2015. Privacy leakage analysis in online social networks. *Computers & Security*, 49 (2015), 239–254. (cited on page 50)

Lipford, H. R.; Besmer, A.; and Watson, J., 2008. Understanding privacy settings in facebook with an audience view. *UPSEC*, 8 (2008), 1–8. (cited on page 1)

Liu, Q.; Chen, Y.; Chen, B.; Lou, J.-G.; Chen, Z.; Zhou, B.; and Zhang, D., 2020. You impress me: Dialogue generation via mutual persona perception. *arXiv preprint arXiv:2004.05388*, (2020). (cited on pages 2, 28, and 29)

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B., 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *2021 IEEE International Conference on Computer Vision (ICCV)*. (cited on page 9)

Madaan, A.; Setlur, A.; Parekh, T.; Poczos, B.; Neubig, G.; Yang, Y.; Salakhutdinov, R.; Black, A. W.; and Prabhumoye, S. Politeness transfer: A tag and generate approach. (cited on page 17)

Manning, C. D.; Raghavan, P.; and Schütze, H., 2008. *Introduction to information retrieval*. Cambridge university press. (cited on page 25)

Martins, A. and Astudillo, R., 2016. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *International Conference on Machine Learning*, 1614–1623. (cited on page 25)

Mathews, A.; Xie, L.; and He, X., 2016. Senticap: Generating image descriptions with sentiments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30. (cited on page 16)

McHugh, M. L., 2012. Interrater reliability: the kappa statistic. *Biochemia Medica*, 22, 3 (2012), 276–282. (cited on page 41)

Medlock, B., 2006. An introduction to nlp-based textual anonymisation. In *LREC*, 1051–1056. Citeseer. (cited on pages xix and 11)

Mehrabi, N.; Morstatter, F.; Saxena, N.; Lerman, K.; and Galstyan, A., 2021. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54, 6 (2021), 1–35. (cited on page 9)

Microsoft, 2021. *Presidio: Data Protection and Anonymization SDK*. https://microsoft.github.io/presidio/. (cited on page 11)

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; and Dean, J., 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems-Volume 2*, 3111–3119. (cited on page 30)

Nallapati, R.; Zhou, B.; dos Santos, C.; Gulçehre, Ç.; and Xiang, B., 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, 280–290. (cited on page 16)

Niu, X.; Rao, S.; and Carpuat, M., 2018. Multi-task neural models for translating between styles within and across languages. In *Proceedings of the 27th International Conference on Computational Linguistics*, 1008–1021. (cited on page 17)

Norberg, P. A.; Horne, D. R.; and Horne, D. A., 2007. The privacy paradox: Personal information disclosure intentions versus behaviors. *Journal of consumer affairs*, 41, 1 (2007), 100–126. (cited on page 1)

Oosterhoff, J. and van Zwet, W. R., 2012. A note on contiguity and hellinger distance. In *Selected Works of Willem van Zwet*, 63–72. Springer. (cited on page 26)

Pan, X.; Zhang, M.; Ji, S.; and Yang, M., 2020. Privacy risks of general-purpose language models. In *2020 IEEE Symposium on Security and Privacy (SP)*, 1314–1331. IEEE. (cited on page 3)

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J., 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 311–318. (cited on pages 40 and 57)

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al., 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12, Oct (2011), 2825–2830. (cited on page 41)

Prabhumoye, S.; Tsvetkov, Y.; Salakhutdinov, R.; and Black, A. W., 2018a. Style transfer through back-translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 866–876. (cited on pages 17 and 36)

Prabhumoye, S.; Tsvetkov, Y.; Salakhutdinov, R.; and Black, A. W., 2018b. Style transfer through back-translation. *arXiv preprint arXiv:1804.09000*, (2018). (cited on page 35)

Qiu, X.; Sun, T.; Xu, Y.; Shao, Y.; Dai, N.; and Huang, X., 2020. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, (2020), 1–26. (cited on page 13)

Radford, A.; Narasimhan, K.; Salimans, T.; and Sutskever, I., 2018. Improving language understanding by generative pre-training. (2018). (cited on pages 15 and 16)

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I., 2019. Language models are unsupervised multitask learners. (2019). (cited on page 57)

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J., 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21 (2020), 1–67. (cited on page 57)

Rao, S. and Tetreault, J., 2018. Dear sir or madam, may i introduce the gyafc dataset: Corpus, benchmarks and metrics for formality style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 129–140. (cited on pages 15 and 16)

Reddy, S. and Knight, K., 2016. Obfuscating gender in social media writing. In *Proceedings of the First Workshop on NLP and Computational Social Science*, 17–26. (cited on pages 38 and 39)

Robert, L. P.; Pierce, C.; Marquis, L.; Kim, S.; and Alahmad, R., 2020. Designing fair ai for managing employees in organizations: a review, critique, and design agenda. *Human–Computer Interaction*, 35, 5-6 (2020), 545–575. (cited on page 7)

Robertson, S. and Zaragoza, H., 2009. *The probabilistic relevance framework: BM25 and beyond*. Now Publishers Inc. (cited on page 25)

Rush, A. M.; Chopra, S.; and Weston, J., 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 379–389. (cited on page 16)

Schein, A. I.; Popescul, A.; Ungar, L. H.; and Pennock, D. M., 2002. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, 253–260. (cited on page 64)

Schofield, C. B. P. and Joinson, A. N., 2008. Privacy, trust, and disclosure online. *Psychological aspects of cyberspace: Theory, research, applications*, (2008), 13–31. (cited on pages 1 and 7)

Shen, L.; Ji, S.; Zhang, X.; Li, J.; Chen, J.; Shi, J.; Fang, C.; Yin, J.; and Wang, T., 2021. Backdoor pre-trained models can transfer to all. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 3141–3158. (cited on page 3)

Shen, T.; Ott, M.; Auli, M.; and Ranzato, M., 2019. Mixture models for diverse machine translation: Tricks of the trade. (cited on page 57)

Sherer, J. A.; Hoffman, T. M.; and Ortiz, E. E., 2015. Merger and acquisition due diligence: a proposed framework to incorporate data privacy, information security, e-discovery, and information governance into due diligence practices. *Richmond Journal of Law & Technology*, 21, 2 (2015), 5. (cited on page 2)

Shi, W.; Cui, A.; Li, E.; Jia, R.; and Yu, Z., 2021. Selective differential privacy for language modeling. *arXiv preprint arXiv:2108.12944*, (2021). (cited on page 8)

Shokri, R.; Stronati, M.; Song, C.; and Shmatikov, V., 2017. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, 3–18. IEEE. (cited on page 8)

Speer, R.; Chin, J.; and Havasi, C., 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31. (cited on pages 55 and 59)

Strengers, Y.; Qu, L.; Xu, Q.; and Knibbe, J., 2020. Adhering, steering, and queering: Treatment of gender in natural language generation. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 1–14. (cited on pages 4, 35, 49, and 50)

Su, J.; Xu, J.; Qiu, X.; and Huang, X., 2018. Incorporating discriminator in sentence generation: a gibbs sampling method. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*. (cited on page 18)

Sudhakar, A.; Upadhyay, B.; and Maheswaran, A., 2019. "transforming" delete, retrieve, generate approach for controlled text style transfer. 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)*, 3260–3270. (cited on page 17)

Sutskever, I.; Vinyals, O.; and Le, Q. V., 2014. Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems*, 27 (2014), 3104–3112. (cited on page 57)

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z., 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2818–2826. (cited on page 37)

Taal, A.; Le, J.; de Leon, A. P.; Sherer, J. A.; and Jenson, K. S., 2017. Technological and information governance approaches to data loss and leakage mitigation. *Comput. Sci. Inf. Technol.*, 5, 1 (2017), 1–7. (cited on page 2)

Tiedemann, J., 2012. Parallel data, tools and interfaces in opus. In *Eight International Conference on Language Resources and Evaluation, MAY 21-27, 2012, Istanbul, Turkey*, 2214–2218. (cited on page 16)

Titus, R. M. and Gover, A. R., 2001. Personal fraud: The victims and the scams. *Crime Prevention Studies*, 12 (2001), 133–152. (cited on page 2)

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I., 2017. Attention is all you need. In *Proceedings of Advances in Neural Information Processing Systems*, 5998–6008. (cited on pages 15, 36, 38, and 39)

Voigt, R.; Jurgens, D.; Prabhakaran, V.; Jurafsky, D.; and Tsvetkov, Y., 2018. Rtgender: A corpus for studying differential responses to gender. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*. (cited on pages 38 and 39)

Wang, A.; Pruksachatkun, Y.; Nangia, N.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. R., 2019. Superglue: a stickier benchmark for general-purpose language understanding systems. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 3266–3280. (cited on page 3)

Weggenmann, B. and Kerschbaum, F., 2018. Syntf: Synthetic and differentially private term frequency vectors for privacy-preserving text mining. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 305–314. (cited on page 11)

Wolf, T.; Sanh, V.; Chaumond, J.; and Delangue, C., 2019. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149*, (2019). (cited on page 57)

Wu, Y.; Schuster, M.; Chen, Z.; Le, Q. V.; Norouzi, M.; Macherey, W.; Krikun, M.; Cao, Y.; Gao, Q.; Macherey, K.; et al., 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, (2016). (cited on pages 40 and 42)

Xu, Q.; Qu, L.; Gao, Z.; and Haffari, G., 2020. Personal information leakage detection in conversations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 6567–6580. (cited on pages 51 and 53)

Xu, Q.; Xu, C.; and Qu, L., 2021. Privacy monitoring service for conversations. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, 1093–1096. (cited on page 13)

Xu, Q. and Zhao, H., 2012. Using deep linguistic features for finding deceptive opinion spam. *Proceedings of the 24th International Conference on Computational Linguistics*, (2012), 1341–1350. (cited on page 36)

Xu, W.; Napoles, C.; Pavlick, E.; Chen, Q.; and Callison-Burch, C., 2016a. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4 (2016), 401–415. (cited on page 15)

Xu, W.; Napoles, C.; Pavlick, E.; Chen, Q.; and Callison-Burch, C., 2016b. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4 (2016), 401–415. (cited on page 57)

Xu, W.; Ritter, A.; Dolan, W. B.; Grishman, R.; and Cherry, C., 2012. Paraphrasing for style. In *Proceedings of the 24th International Conference on Computational Linguistics*. (cited on page 16)

Yan, Y.; Qi, W.; Gong, Y.; Liu, D.; Duan, N.; Chen, J.; Zhang, R.; and Zhou, M., 2020. Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. *arXiv preprint arXiv:2001.04063*, (2020). (cited on page 55)

Yang, Y.; Zhang, Y.; Tar, C.; and Baldridge, J., 2019. Paws-x: A cross-lingual adversarial dataset for paraphrase identification. (cited on page 57)

Zhang, B. H.; Lemoine, B.; and Mitchell, M., 2018a. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 335–340. (cited on pages xvii and 10)

Zhang, S.; Dinan, E.; Urbanek, J.; Szlam, A.; Kiela, D.; and Weston, J., 2018b. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th ACL*, 2204–2213. (cited on pages 2, 12, 21, 22, 28, and 29)

Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K. Q.; and Artzi, Y., 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*. (cited on page 57)