12-11-2023

Privacy-Preserving Deep Learning Mechanisms for Multimedia Data-Oriented Applications

Honghui Xu Ph.D.
Georgia State University

Privacy-Preserving Deep Learning Mechanisms for Multimedia Data-Oriented Applications

by

Honghui Xu

Under the Direction of Zhipeng Cai, Ph.D.

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2023

ABSTRACT

Nowadays, with the proliferation of multimedia and the coming of deep learning era, many multimedia data-oriented applications have been proposed to achieve face recognition, automatic retailing, automatic driving, intelligent medical healthcare, visual-audio speech recognition, and so on. However, these deep learning models may face a serious risk of data privacy leakage in the utilization process of these multimedia data. For example, malicious attackers can exploit deep learning techniques to deduce sensitive information from eavesdropped multimedia data, and these attackers can pilfer historical training data through a membership inference attack. Although some privacy-preserving deep learning approaches have been investigated, there are many limitations to be overcome. So far, it is still an open issue to design privacy-preserving deep learning mechanisms in different application scenarios to achieve individuals' privacy protection while maintaining deep learning models' performance.

In this dissertation, we investigate a series of mechanisms for multimedia data privacy protection in deep learning applications. Firstly, we propose an audio-visual autoencoding scheme to achieve visual privacy protection, visual quality preservation, and video transmission efficiency. Secondly, we propose a differential private deep learning model to realize the tradeoff between data privacy and the utility of multi-label image recognition (*e.g.*, accuracy) by leveraging a differential privacy mechanism with a bounded global sensitivity and incorporation of regularization term into loss function. Thirdly, we propose a differential private correlated representation learning model to accomplish privacy-preserving multimodal sentiment analysis by combining a correlated representation learning scheme with a differential privacy protection scheme. Especially, a pre-determined correlation factor is employed to flexibly adjust the expected correlation among the correlated representations.

At last, we also propose the future research topics to complete the whole dissertation.

The first topic focuses on the multi-sensor data privacy protection while considering the certified performance of deep learning. The second topic studies model privacy protection to prevent side-channel attacks from inferring the architecture of deep neural networks.

INDEX WORDS: Visual Data Privacy, Multimodal Data Privacy, Differential Privacy, Deep Learning, Sensitive Information

Privacy-Preserving Deep Learning Mechanisms for Multimedia Data-Oriented Applications

by

Honghui Xu

Committee Chair:	Zhipeng Cai
Committee:	Zhipeng Cai
	Yingshu Li
	Wei Li
	Yan Huang

Electronic Version Approved:

Office of Graduate Services

College of Arts and Sciences

Georgia State University

December 2023

DEDICATION

To my parents, Jin Xu and Liping Hong, whose unwavering support and boundless encouragement have been the foundation of my academic journey. Your belief in my dreams made it possible for me to embark on the path of undergraduate and doctoral studies, and for that, I am eternally grateful.

To my friend, Dr. Danyang Zheng, who was a guiding light during the initial years of my PhD journey in the United States. Your invaluable living suggestions and constant friendship eased the transition into a new and unfamiliar environment, making those early days of research and adaptation far more manageable.

I owe a special debt of gratitude to JD. Yan Chen, whose unwavering encouragement and companionship sustained me through the numerous challenges I faced during the latter three years of my doctoral pursuit. Your presence was a source of comfort, and your belief in my abilities kept me motivated when the journey grew arduous.

I extend my heartfelt thanks to all my friends for their steadfast support and camaraderie. Your friendship and encouragement were instrumental in maintaining my enthusiasm and determination throughout my academic pursuits.

Finally, I wish to express my appreciation to all the members of my current research group, whose collaborative efforts and shared passion for knowledge have enriched my academic experience. Together, we have faced research challenges and achieved remarkable milestones, and I am grateful for the collective dedication and contributions of each team member.

This dedication is a tribute to the cherished individuals who have played vital roles in my educational and personal growth. Your unwavering support and companionship have made my academic journey a meaningful and rewarding endeavor.

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my advisor, Dr. Zhipeng Cai, for his unwavering support and invaluable contributions to my academic journey. Dr. Cai has been a pillar of guidance, providing me with essential research suggestions. His mentorship of both junior Ph.D. students and undergraduate students has been a source of inspiration and knowledge that I will carry with me throughout my career. Dr. Cai's dedication to fostering academic growth has been instrumental in shaping my dissertation, and I am truly fortunate to have had the privilege of working under his guidance.

I am equally indebted to my co-advisor, Dr. Wei Li, for her continuous support and expert guidance throughout my research endeavors. Her insightful suggestions and unwavering commitment to my success have been integral to the development of my dissertation. Dr. Li's guidance has been a beacon of light in navigating the complexities of my research, and I am truly grateful for her contributions.

I extend my sincere appreciation to the members of my committee, Dr. Yingshu Li and Dr. Yan Huang, for their valuable insights and feedback, which have significantly enriched the quality of my dissertation. Their constructive suggestions and critical evaluations have been essential in shaping my research and ensuring its rigor.

I would also like to acknowledge Dr. Yong Deng, Dr. Ling Tian, and Dr. Zhao Kang for their instrumental contributions during my undergraduate studies. Their mentorship and insightful suggestions motivated me to pursue a Ph.D. and have been a source of constant

encouragement throughout my academic journey. I am deeply grateful for the lasting impact they have had on my educational and professional development.

In summary, I am profoundly grateful for the collaborative spirit and the unwavering support of all these individuals, who have collectively played a vital role in helping me shape my colorful academic life. Their guidance, mentorship, and contributions have been invaluable in my pursuit of knowledge and academic excellence.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

Multimedia data-oriented deep learning applications refer to the use of deep learning techniques to analyze and process large amounts of multimedia data, such as images, videos, and audio recordings. These applications are designed to extract meaningful information and insights from multimedia data, and can be used in a variety of fields, including social media, finance, , healthcare, and more. For example, in social media, deep learning can be used to improve the quality of video and audio sharing. In finance, deep learning can be used to analyze stock market data and make predictions about future trends. In healthcare, deep learning algorithms can be used to analyze medical images and help identify early signs of disease. Overall, multimedia data-oriented deep learning applications have the potential to revolutionize the way we process and understand large volumes of multimedia data, leading to new insights and discoveries across a wide range of fields.

However, privacy leakage is a significant concern when using multimedia data-oriented deep learning applications. These applications often require large amounts of data, including personal and sensitive information, to train and improve their algorithms. As a result, there is a risk that this data can be leaked or hacked, leading to privacy breaches and potential harm to individuals. For example, if a healthcare organization uses deep learning to analyze medical images, and that data is leaked, patients' personal health information could be compromised. Similarly, if a company uses deep learning to analyze customer data, and that data is hacked, customers' personal and financial information could be exposed.

Overall speaking, privacy leakage is a critical concern in the use of multimedia data-oriented deep learning applications, and must be carefully managed to protect individuals and organizations from harm. Although some privacy-preserving deep learning approaches have been proposed, these previously published models still have some limitations. Therefore, it is still worthy of investigating privacy-preserving deep learning models according to different application scenarios.

There are two common deep learning application scenarios, including end-to-end deep learning applications and third-party deep learning applications.

The end-to-end deep learning application scenario refers to a specialized approach within the field of artificial intelligence, where a single neural network model is designed and trained to perform a specific task. In this scenario, the raw input data is transmitted from the users' side to the server's side for further prediction. End-to-end deep learning is particularly well-suited for tasks such as image and speech recognition, natural language processing, and autonomous driving, where the model learns to extract relevant features and make decisions directly from raw data.

In a third-party deep learning application scenario, external organizations or developers leverage pre-trained deep learning models and services provided by a third party to enhance their applications. This approach involves integrating specialized AI capabilities, such as image recognition, language processing, or recommendation systems, into their software, without the need to develop and train these models from scratch. By incorporating third-party deep learning services, businesses, and individuals can rapidly access and deploy cutting-edge AI technologies, significantly reducing the development time and costs associated with implementing complex machine learning solutions. This scenario is commonly used in various industries, including e-commerce, healthcare, and content recommendation systems, where integrating advanced AI capabilities can provide a competitive edge and improve user experiences.

In addition, there are three mainstream data privacy leakage ways in deep learning application scenarios, consisting of eavesdropping attacks, side-channel information attacks, and membership inference attack.

Eavesdropping attacks in deep learning applications represent a security threat where malicious actors intercept or gain unauthorized access to sensitive information during data transmission. Eavesdropping can occur in various contexts, such as unencrypted communication channels. As deep learning systems become more prevalent in critical domains like healthcare, finance, and autonomous vehicles, safeguarding against eavesdropping attacks is essential to protect user privacy and intellectual property.

Side-channel information attacks in deep learning applications are a sophisticated class of security threats wherein adversaries exploit unintended information leakage from a system's physical or computational side channels to infer sensitive data. By carefully analyzing these side-channel signals, attackers can uncover confidential information. As deep learning models are increasingly deployed in various real-world applications, protecting against side-channel information attacks has become a critical concern.

The membership inference attack in the context of deep learning applications is a privacy breach where an adversary attempts to determine whether a specific data point was part of the training dataset used to build a machine learning model. This attack is particularly concerning in scenarios where the training data contains sensitive or personal information. Membership inference attacks are especially relevant in applications like healthcare and finance, where the disclosure of data sources can have legal, ethical, and security implications.

In this dissertation, we investigate three multimedia data privacy protection schemes in deep learning application scenarios by considering three aspects, including specific application scenarios, privacy leakage ways, and data characteristics.

In the first part, we study a visual privacy protection problem during video streaming transmission in the end-to-end deep learning applications. We propose a cycle vector-quantized variational autoencoder framework to defend eavasdropping attack, in which a fusion mechanism is designed to integrate the video and its extracted audio. The extracted audio works as the random noise with a non-patterned distribution, which outperforms the noise that follows a patterned distribution for hiding visual information in the video. Moreover, the video streaming is compressed by taking into account temporal correlation in video in the encoding process of the proposed framework, which can resist side-channel information attack during video transmission and reduce video transmission time.

In the second part, we study a training data privacy protection problem in the third-party deep learning applications. We design a differential private deep learning model by implementing differential privacy mechanism on the model's outputs to defend membership inference attack and avoid large aggregated noise simultaneously. Meanwhile, a regularization term is exploited in the loss function to increase the model prediction accuracy and robustness, and a bounded global sensitivity in differential privacy is used to mitigate excessive noise' side effect and obtain a performance improvement. Theoretical proof shows that our proposed model can guarantee differential privacy for model's outputs, weights and inputs while preserving model robustness.

In the third part, we study a multimodal data privacy protection problem in the end-to-end deep learning applications. We create a differential private correlated representation learning model to realize privacy-preserving multimodal sentiment analysis by combining a correlated representation learning scheme with a differential privacy protection scheme. Our correlated representation learning scheme aims to achieve heterogeneous multimodal data transformation to meet the requirements of privacy-preserving multimodal sentiment analysis. The differential privacy protection scheme is used to obtain the disturbed correlated and uncorrelated representations by adding Laplace noise for ϵ-differential privacy. Particularly, the correlation factor can flexibly adjust the expected correlation among the correlated representations and help alleviate the side-effect of the added Laplace noise on the sentiment prediction performance.

Finally, we provide a concise overview of our forthcoming work, which serves as the final component of our comprehensive dissertation. The first area of focus centers on enhancing multi-sensor data privacy protection, with a special emphasis on preserving the certified performance of deep learning models. The second facet of our research investigates techniques for safeguarding model privacy to counteract side-channel attacks that attempt to deduce the architecture of deep neural networks. This section is dedicated to addressing the emerging challenges arising from the privacy leakage associated with multi-sensor data and model architectures in deep learning applications.

CHAPTER 2

AUDIO-VISUAL AUTOENCODING FOR PRIVACY-PRESERVING VIDEO STREAMING

2.1 Motivation

Recently, sharing video streaming has been becoming increasingly popular with the wide applications of Internet of Things (IoT) devices Wu et al. (2019a); Verma (2020); Onohara et al. (2019); Ong et al. (2019); Wang et al. (2023a); Xu et al. (2022a); Chi et al. (2021), the number of which is predicted to reach about 45 billion by 2022 Capital (2017); Zheng et al. (2018a). In transmission process, however, the video streaming may be maliciously intercepted by attackers who intend to infer individuals' private information from the videos using detection/prediction approaches Li et al. (2019a); Ulutan et al. (2020); Liu et al. (2019); Wu et al. (2019b); Li et al. (2016a); Jiang et al. (2020); Anderson et al. (2019); Huang et al. (2009); Xiong et al. (2022); Cai et al. (2016); Liang et al. (2018). Meanwhile, recent breakthroughs in deep learning accelerate the development of machine learning-based detection techniques Xu et al. (2023b), such as face detection Nasir et al. (2019); Zhang et al. (2020, 2019a); Li et al. (2019b, 2020a) and semantic segmentation Zheng et al. (2015); Ghanem et al. (2019); Meenpal et al. (2019); Benini et al. (2019); Wang et al. (2019c), which greatly increases the risk of privacy leakage in the video streaming Zheng et al. (2020b, 2018b); Cai & Zheng (2018). For example, from the video, attackers are able to use these advanced machine learning models to accomplish speech recognition Hung & Ba (2009); Chaudhuri et al. (2018), action recognition Gao et al. (2019); Roth et al. (2019), and other activity detection. According to the latest Cost of a Data Breach Report proposed by IBM and the Ponemon Institute, privacy leakage causes property loss of millions dollars every year for individuals or companies concerned IBM & the Ponemon Institute (2019); Cai & He (2019). In addition, privacy protection has been regulated by law – on May 25th, 2018, the European Union's new General Data Protection Regulation (GDPR) came into force, requiring that people should have more control over their personal data. To this end, *privacy protection is deemed to be an indispensable component for video sharing.*

So far, a lot of research has been conducted to protect visual privacy in various ways. Some works aim to hide (partial) visual information for privacy protection Brkić et al. (2017); Uittenbogaard et al. (2019); Mirjalili et al. (2018); Xiong et al. (2020, 2019); **?**, some approaches achieve anonymity through disturbing the original visual information Meng et al. (2019); Tang et al. (2017); Kim & Yang (2019); Wang et al. (2019a); Cai et al. (2019), some methods protect privacy by changing the visual style of original information Wu et al. (2019a); Chen et al. (2018), and some studies apply encryption methods to protect privacy in video Paruchuri et al. (2009); Liu & Kong (2018); Zhang et al. (2012, 2010); Chu et al. (2013). However, *the existing works still have their limitations, which also challenges the design of effective protection for visual privacy*: (i) random noise is added to disturb the visual information in noise-based models, but the added noise usually follows some patterned distributions (*e.g.*, normal distribution), which can be utilized as prior knowledge in attackers' detection models to infer private information; (ii) some noise-based models are just trained to fool a certain kind of discriminative model, which cannot be used to defend general detection models in real applications; (iii) all the existing models, even the encryption-based ones, do not fully consider leakage of side-channel information (*e.g.*, traffic size) during video transmission, leading to vulnerability to side-channel inference attack; and (iv) these previous privacy-preserving models only focus on visual privacy in separated video frames but overlook the temporal information (*i.e.* the relations between frames) in video streaming, resulting in the low effect of privacy protection.

To overcome the above challenges, in this paper, *we propose to encode and decode video streaming with its extracted audio to achieve visual privacy protection while maintaining the expected visual quality and enhancing video transmission efficiency.* The extracted audio is a kind of random noise without any patterned distribution, which can better disturb the visual information as well as reduce the accuracy of malicious detection, compared with the noise that follows patterned distributions. For any video, its extracted audio cannot be generated or manipulated easily by attackers without any prior knowledge, which ensures that the encoded video can only be decoded by the receivers who obtain the extracted audio. In other words, we aim to fuse multiple heterogeneous data sources (*i.e.*, the video and its extracted audio in this paper) to hide private visual information to defend detection attack and side-channel inference attack simultaneously during video transmission, which has not been addressed in literature.

To realize our proposed design, we develop a cycle-VQ-VAE framework to accomplish the fusion of heterogeneous data sources by employing the idea of codebook. Our framework consists of two VQ-VAE components with one working as the encoder and the other working as the decoder. Considering a pair of sender and receiver in video sharing applications, this kind of cycle framework can guarantee that the encoded video frame can be properly encoded at the sender and decoded at the receiver with high visual quality. To fuse different data sources, we map both the video and its extracted audio into an appropriate low-dimension space such that the codes of audio can disturb the codebook of video and the video information can be compressed effectively in the encoder. This encoding process that has not been presented in previous works makes sure that our framework can also be used to defend side-channel inference attack because it changes the traffic pattern of video streaming. Correspondingly, in the decoder, the same audio can be used to decode the encoded video by removing the extra codes of the audio from the disturbed codebook. Under this cycle-VQ-VAE framework, we develop two different models, including Frame-to-Frame (F2F) and Video-to-Video (V2V) models. In F2F model, we divide the video into a series of frames and reconstruct the images in a frame by frame manner. In V2V model, we treat the video as time-series data to perform image reconstruction taking into account the temporal information in video. Finally, we use the AVE dataset Gu et al. (2018a), two AI detection models, and one side-channel inference attack model to evaluate the superiority of our proposed F2F and V2V models over the state-of-the-art schemes in terms of visual privacy protection, visual quality preservation, and video transmission efficiency. In the following, the contributions of this paper are summarized.

- To the best of our knowledge, this is the first work to study the fusion of multiple heterogeneous data sources in video streaming for privacy protection.

- The extracted audio used in the cycle-VQ-VAE framework does not follow any patterned distribution and thus outperforms the works using the noise that follows some patterned distributions (*e.g.*, normal distribution).

- A novel cycle-VQ-VAE framework is developed to process video streaming, where the video and its extracted audio can be fused properly for protecting visual privacy, preserving visual quality, and compressing video information simultaneously.

- The integration of video compression and encoding is proposed to defend side-channel inference attack and reduce video transmission overhead.

- F2F and V2V models are designed under the cycle-VQ-VAE framework to achieve the goal of privacy protection; especially, V2V model exploits the temporal information for performance enhancement in privacy protection, video compression, and video reconstruction.

- The real-data experiment results confirm the effectiveness and the advantages of our proposed models compared with the state-of-the-art.

The rest of this paper is organized as follows. Related works are briefly summarized in Section 2.2. After introducing preliminaries in Section 2.3, we detail our proposed models in Section 2.4. In Section 2.5, comprehensive experiments are conducted and analyzed. Finally, Section 2.6 concludes this paper and discusses our future work.

2.2 Related Works

The state-of-the-art about visual privacy protection is summarized in this section.

2.2.1 Noise-based Privacy-Preserving Models

In the existing works, the methods of protecting visual privacy via adding noise can be classified into three main categories: (i) applying noise to disturb the feature attributes in order to decrease the accuracy of recognition results Brkić et al. (2017); Uittenbogaard et al. (2019); Mirjalili et al. (2018); (ii) using steganography algorithms to generate the stego images to protect privacy Meng et al. (2019); Tang et al. (2017); Kim & Yang (2019); and (iii) changing the image styles to hide original visual information for privacy preservation Wu et al. (2019a); Chen et al. (2018).

Raval et al. (2017) designed a perturbation mechanism that can obtain the trade-off between privacy and utility to protect visual secrets based on denoising autoencoder through the adversarial training. Brkić et al. (2017) proposed to hide some biometric attributes with noise to reduce the accuracy of face recognition. They also proposed a Conditional Generative Adversarial Network (CGAN) to generate a human image of full body while offering a solid level of identity protection in Brkic et al. (2017). Uittenbogaard et al. (2019) designed a framework based on Generative Adversarial Network (GAN) to achieve the goal of detecting, removing, and inpainting moving objects in multi-view imagery while removing private regions that users care about. Meng et al. (2019) proposed a steganography algorithm based on image-to-image translation using cycle-GAN to obtain the stego images for the purse of concealment and security in the transmission process. Tang et al. (2017) developed an automatic stegangraphic distortion framework using GAN (named ASDL-GAN), which can be applied to images for the enhancement of privacy preservation. Kim & Yang (2019) proposed a privacy-preserving adversarial protector network (termed PPAPNet), where a noise amplifier was used to optimize noise for effective image anonymization. Wu et al. (2019a) designed a method to keep video transmission secure by using a two-dimensional noise matrix as the 4-th channel of image combining with a 3-channel RGB image, in which a video frame was transformed from one style to another based on the architecture of cycle-GAN. Chen et al. (2018) also proposed to transfer the realistic images into cartoon images based on GAN to protect privacy to a certain extent.

2.2.2 Encryption-based Privacy-Preserving Models

Besides, encryption-based methods are proposed to hide the private visual information in video.

Paruchuri et al. (2009) encrypted foreground video bit-stream to hide the private information in surveillance systems. Liu & Kong (2018) obscured the human face region in real time by encrypting the spatial chaotic map of face. Zhang et al. (2010) generated a key through a cryptographic MAC function by using the information of the head contour in the video frame, and the key is used in a stream cipher to lock the head information detected pedestrians for privacy preservation. Chu et al. (2013) proposed a fast homomorphic encryption method to encrypt the video frames for secure video transmission.

2.2.3 Limitations of Existing Works

In the existing noise-based models, the used noise follows the normal distribution, which, however, can be utilized as prior knowledge by attackers to mitigate the impact of noise in their detection models and enhance the accuracy of information prediction. Even for the encryption-based models, all of the current works fail to fully consider privacy leakage in the video transmission process and thus may be vulnerable to the side-channel inference attack where attackers are able to infer private information by analyzing the users' traffic data Li et al. (2016a). What's worse, recent advanced machine learning models can achieve action recognition and activity detection in video by exploiting the temporal information (*i.e.* the relations between frames) Jiang et al. (2020); Gao et al. (2019); Roth et al. (2019), which has not been taken into account for privacy preservation yet. Due to the aforementioned limitations, these existing works may not be adequate to effectively accomplish the task of protecting visual privacy in video.

In this paper, to improve the performance of visual privacy protection, we propose F2F and V2V models based on cycle-VQ-VAE to encode and decode the video by employing the video's extracted audio and temporal information. The technical advantages and innovations of our models lie in several aspects. (i) The audio of a video is extracted as the noise whose distribution is random and unknown. Thus, *applying such extracted audio can disturb the visual information more effectively, compared with using the noise following patterned distribution (e.g., normal distribution).* (ii) Different from the noise that follows patterned distribution, the extracted audio is unique and meaningful for its corresponding video, so that *it guarantees that the noise cannot be generated or manipulated easily and can be used to decode the encoded video only by the receivers who have the audio.* (iii) The process of video compression is incorporated into our cycle-VQ-VAE framework, *improving the resistance to side-channel inference attack during transmission and reducing the video transmission time.* (iv) The relations between frames are utilized in V2V by integrating cycle-VQ-VAE with the RNN layers, *making privacy protection, video compression, and video reconstruction more efficient.*

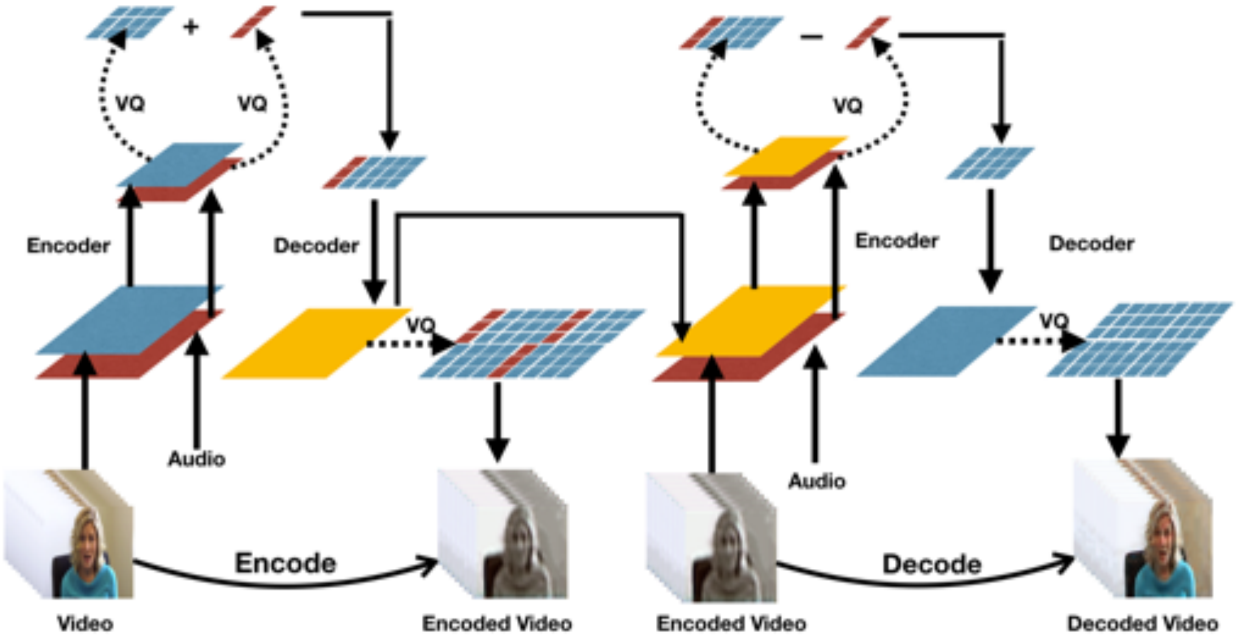

Figure 2.1 The architecture of our cycle-VQ-VAE model

2.3 Preliminaries

Vector Quantized Variational AutoEncoder (VQ-VAE) is a state-of-the-art image generation model with convolutional layers' architecture, in which all features of video frames are mapped into the codebook Razavi et al. (2019). With the help of codebook, high-dimension data can be mapped into a low-dimension space and also can be reconstructed from the mapped low-dimension space.

VQ-VAE model consists of one encoder E and one decoder D, in which E and D share a common codebook c. The encoder is used to embed the original observations x into feature maps that should be close to the codebook vector c, and the decoder is used to recover the original observations $\|x - D(c)\|_2^2$ using the codebook vector c. During this process, performance loss includes: (i) the codebook loss, which is the distance between the selected codebook c and the outputs of encoder and is computed by $\|sg[E(x)] - c\|_2^2$ with the codebook variables, and (ii) the communication loss, which is the distance between the outputs of encoder and the selected codebook c and is calculated via $\|sg[c] - E(x)\|_2^2$ with the encoder weights, where $E(x)$ is the output of the encoder, sg is the stop-gradient to learn the code mappings for the codebook generation, and β is a hyperparameter to control the reluctance to change the codebook c to the encoder output. The objective function of VQ-VAE is expressed in Eq. (2.1).

$$L = \|x - D(c)\|_2^2 + \|sg[E(x)] - c\|_2^2 + \beta\|sg[c] - E(x)\|_2^2. \tag{2.1}$$

2.4 Methodology

In this section, we propose a cycle Vector Quantized Variational AutoEncoder (cycle-VQ-VAE) framework, based on which we design two novel models to generate privacy-preserving video.

2.4.1 Cycle-VQ-VAE Framework

The architecture of our cycle-VQ-VAE framework is shown in Fig. 2.1. This framework consists of one encoder and one decoder, where the encoder is designed to generate the encoded video frames for privacy protection, the decoder is designed to recover the encoded video frames, and the process of mapping video is based on VQ-VAE.

In the encoder of our cycle-VQ-VAE framework, the video frames and its extracted audio that are of high-dimension data are mapped into a low-dimension space. The low-dimension representations of the audio are treated as the extra codes and added into the original codebook of video frames. Then, the disturbed codebook is used to generate the encoded video for privacy-preserving transmission. In the decoder, the low-dimension representations of the audio are removed from the disturbed codebook, and the original video frames can be reconstructed from the clean codebook.

It is worth mentioning that mapping high-dimension data into a low-dimension space is not a trivial issue. If the information in the codebook of video frames is much more than that in the codebook of audio in the low-dimension space, the codes of audio are not enough to disturb the codebook of video frames; if the information in the codebook of video frames is much less than that in the codebook of audio in the low-dimension space, it will be hard to extract the extra codes from the disturbed codebook of video frames to reconstruct the original video frames. That is, it is necessary to explore an appropriate low-dimension space, in which the codebook of video frames can be effectively disturbed using the codebook of its extracted audio. In this paper, we do comprehensive experiments by adjusting the dimension of codebook in the training process until we find a proper low-dimension space such that the encoded video frame reconstructed by the disturbed codebook is hardly detected by AI detection models, and the decoded video frame reconstructed by the clean codebook is similar to the original video frame.

Under our proposed cycle-VQ-VAE framework, a frame-to-frame (F2F) model and a video-to-video (V2V) model are developed. Especially, by utilizing the relations between video frames, V2V obtains an enhanced performance of privacy protection, video compression, and video reconstruction. The details of F2F and V2V models are demonstrated in Section 2.4.2 and Section 2.4.3, respectively.

2.4.2 Frame-to-Frame (F2F) Model

2.4.2.1 Encoder

The encoder in F2F model includes one encoder module, one decoder module, and one codebook c_a^v as shown in Fig. 2.2. We encode the video frames with its extracted audio a to generate the encoded video v_a for protecting visual privacy. In other words, we use the low-dimension representations of audio as the extra codes c_a to disturb the codebook of the video frames c_v.

In the encoder module, we map both the video frames v and the audio a into the low-dimension space represented by codebook c_a^v, which is performed by using the stop-gradient sg Razavi et al. (2019). Let $E(v_a^v|(v,a))$ be the expectancy of obtaining the encoded video with the video frames and the audio as inputs. According to the VQ-VAE mechanism, we

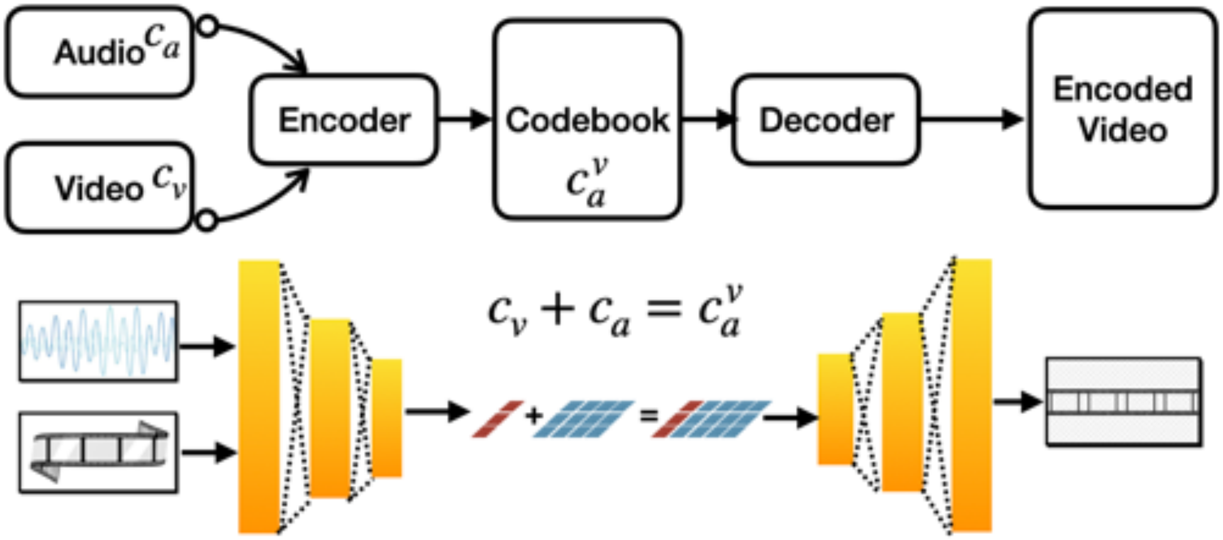

Figure 2.2 The process of encoding (adding c_a into codebook c_a^v)

can compute the codebook loss in Eq. (2.2) and the commitment loss in Eq. (2.3).

$$L_{E1} = \|sg[E(v_a|(v,a))] - c_a^v\|_2^2. \tag{2.2}$$

$$L_{E2} = \|sg[c_a^v] - E(v_a|(v,a))\|_2^2, \tag{2.3}$$

where $||\cdot||_2^2$ denotes the squared L2-norm.

In the decoder module, we generate the encoded video frames v_a from the disturbed codebook c_a^v, in which the reconstruction loss is computed by Eq. (2.4).

$$L_{D1} = \|v_a - D(c_a^v)\|_2^2. \tag{2.4}$$

To sum up, the loss function of the encoder in F2F model can be expressed in Eq. (2.5).

$$L_{Total1} = L_{E1} + \beta_e L_{E2} + L_{D1}, \tag{2.5}$$

where β_e is a hyperparameter to control the reluctance to change the codebook c_a^v to the encoded video v_a.

2.4.2.2 Decoder

At the side of receivers, the encoded video and the audio are high-dimension data. In order to obtain the original video, we first map the received data into the low-dimension space so as to clean the disturbed codebook of encoded video in the low-dimension space. Then, we reconstruct the decoded video in the high-dimension space.

Accordingly, the decoder in F2F model also has three components, including one encoder module, one decoder module, and one codebook c_v as shown in Fig. 2.3. In the decoder, we use the same audio a to decode the encoded video frames v_a with an aim that the decoded video frames should be similar to the original video frames v. To this end, we remove the extra codes c_a from the disturbed codebook c_a^v to obtain the clean codebook c_v of video frames.

In the encoder module, we map both the encoded video frames v_a and the audio a into low-dimension space and learn the mappings through the stop-gradient sg operation. Let $(c_a^v|a)$ denote the disturbed codebook c_a^v, in which the codes of audio a are removed and $E(v|(v_a, a))$ denote the expectancy of obtaining decoded video frames with the encoded video frames and the audio being the inputs. The codebook loss and the commitment loss in this VQ-VAE are calculated by Eq. (2.6) and Eq. (2.7), respectively.

$$L_{E3} = \|sg[E(v|(v_a, a))] - (c_a^v|a)\|_2^2. \tag{2.6}$$

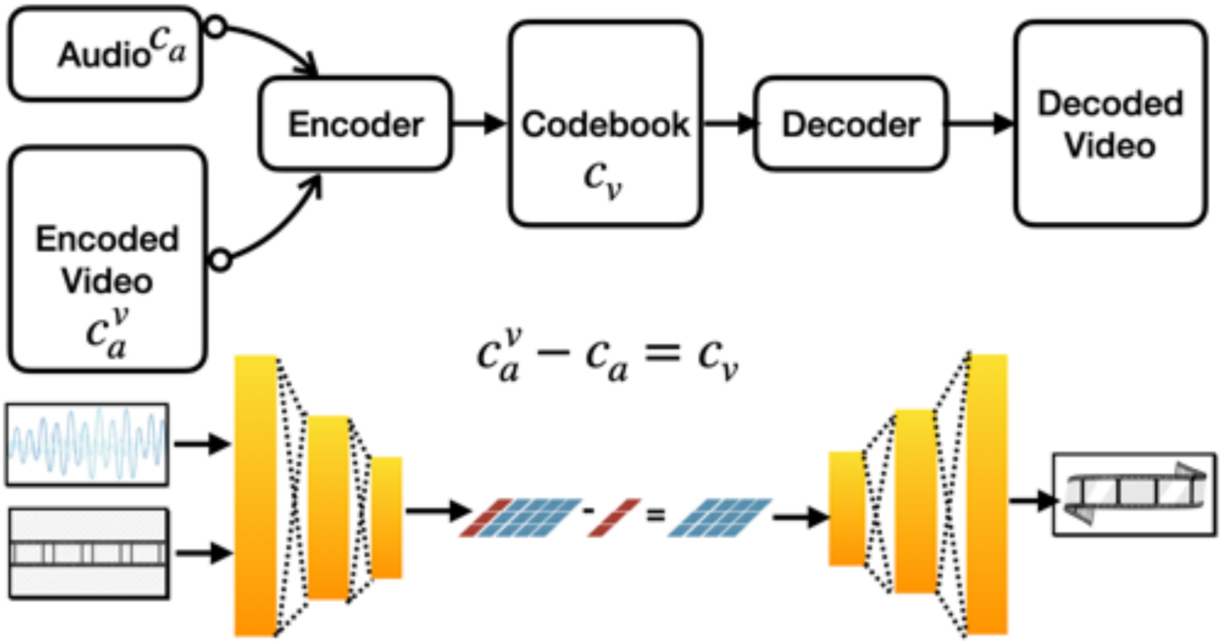

Figure 2.3 The process of decoding (removing c_a from codebook c_a^v)

$$L_{E4} = \|sg[c_a^v|a] - E(v|(v_a, a))\|_2^2. \tag{2.7}$$

In the decoder module, we produce the decoded video frames from the clean codebook such that the decoded video frames are similar to the original video frames v. We remove the codes of audio c_a from the disturbed codebook c_a^v. The reconstruction loss is shown below.

$$L_{D2} = \|v - D(c_a^v|a)\|_2^2. \tag{2.8}$$

The loss function of the decoder in F2F model can be calculated by Eq. (2.9).

$$L_{Total2} = L_{E3} + \beta_d L_{E4} + L_{D2}, \tag{2.9}$$

where β_d is a hyperparameter to control the reluctance to change the clean codebook c_v to the original video v.

In summary, the loss function of our proposed F2F model is as follows,

$$L_{Total} = L_{Total1} + L_{Total2}. \tag{2.10}$$

We aim to minimize Eq. (2.10) in the training process, where L_{Total1} is minimized to obtain the encoded video frames using its extracted audio and L_{Total2} is minimized to decode the encoded video frames using the same audio such that the decoded video is similar to the original video.

2.4.3 Video-to-Video (V2V) Model

In F2F model, we divide the video into a series of frames and reconstruct the images in a frame by frame manner without considering the relations between frames. Motivated by the idea of video reconstruction in Wang et al. (2018, 2019b); Mallya et al. (2020); Chen et al. (2019a), we propose V2V model with the help of RNN layers, in which the temporal information (*i.e.* the relations between frames) in video is used for performance improvement in protection visual privacy, compressing video, and reconstructing video.

The architectures of encoder and decoder in V2V model are presented in Fig. 2.4 and Fig. 2.5, respectively. The difference between our F2F and V2V models is that we deploy a recurrent layer after each Convolutional Neural Network (CNN) block. A hidden state h in each recurrent layer (denoted by function f) is an output from the previous time step, i.e., for i-th CNN block, the output is $o_i = h_i = f(v, h_{i-1})$, where h_i is the hidden state in the i-th CNN block, and h_{i-1} is the hidden state in the $(i-1)$-th CNN block.

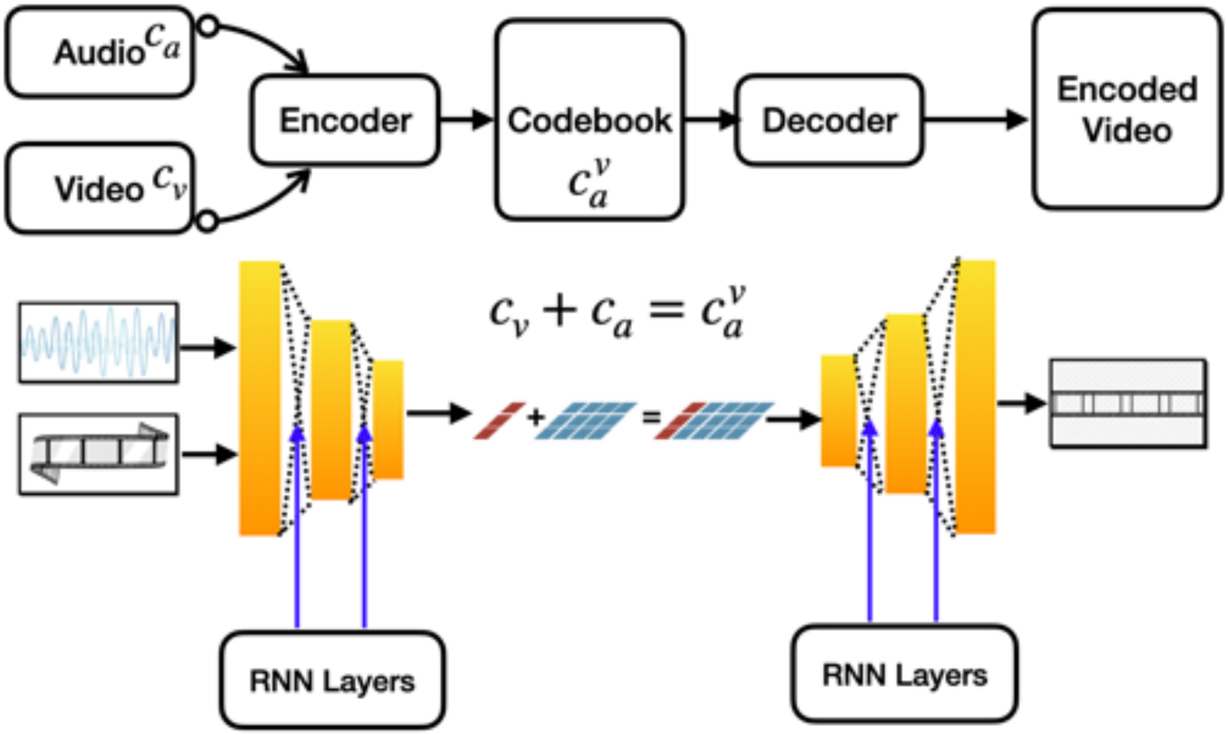

Figure 2.4 The encoder architecture of V2V

2.5 Experiment and Analysis

In order to validate the effectiveness of our F2F and V2V models, extensive experiments are conducted to qualitatively and quantitatively evaluate the results of video encoding/decoding, the performance of privacy protection, and the efficiency of video transmission.

2.5.1 Experiment Settings

2.5.1.1 Dataset

In our experiments, we extract the video frames and the audio from 200 videos in the AVE dataset Gu et al. (2018a) to form the video dataset and audio dataset.

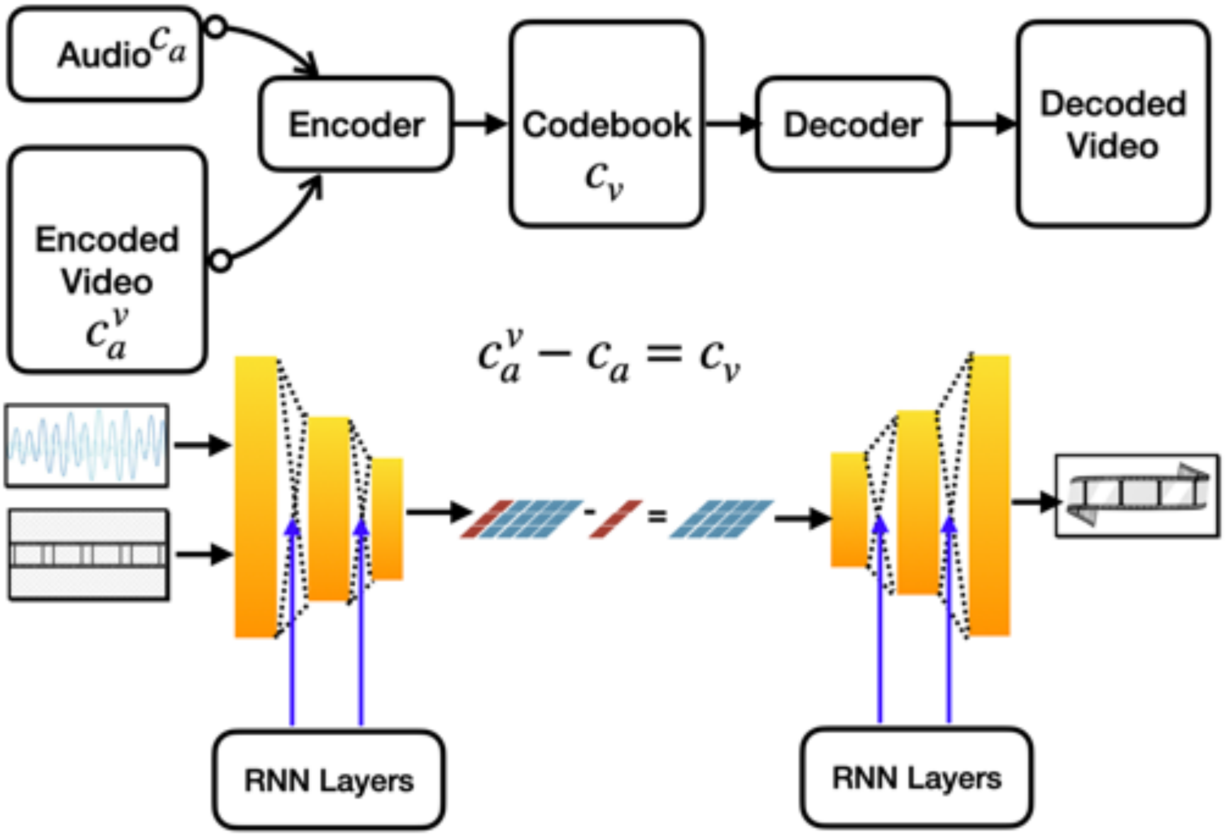

Figure 2.5 The decoder architecture of V2V

2.5.1.2 AI Detection Models for Video Frames

To illustrate that in our F2F and V2V models, the encoded video frames can resist AI detection and the decoded video frames can maintain visual quality, we adopt two AI detection models that have been widely used in real applications with mature technology. One is a face detection model that can detect the human face with a rectangle Nasir et al. (2019), and the other is the semantic segmentation model that can segment the human body with a pink color Zheng et al. (2015).

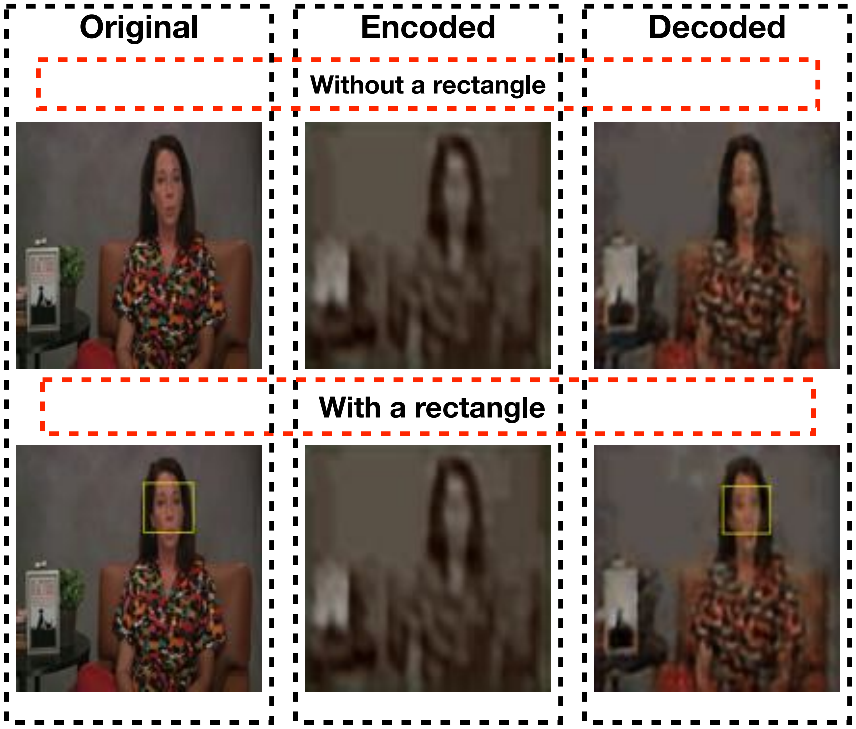

Figure 2.6 Face Detection on F2F Video Frames

2.5.1.3 Side-Channel Inference Attack Model for Video Streaming

In real applications, a video can be typically encoded via a standard encoding method H264 Grecos & Yang (2005) and then encrypted by TLS/SSL using 128-bit AES Lee et al. (2007) for secure transmission. Nevertherless, the traffic pattern can be still utilized as side-channel information to infer individuals' activities in video streaming as the data traffic size can indicate the existence/type of an activity, resulting in privacy leakage. In our experiments, the attack approach of Li et al. (2016a) is adopted, in which the traffic streaming is firstly divided into separate parts and then statistical coefficients (including mean, variance, skewness and kurtosis) of each separated traffic data are used as features to do activity recognition by using k-NN classification algorithm.

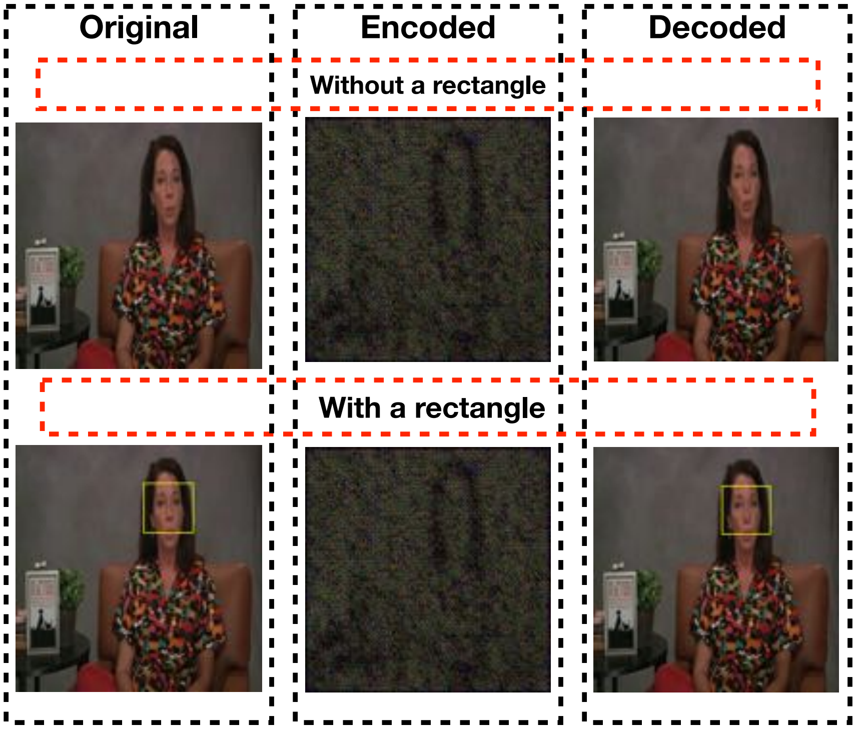

Figure 2.7 Face Detection on V2V Video Frames

2.5.1.4 Two Baselines

We compare our proposed F2F and V2V models with two baselines. (1) AE based model: it is based on autoencoder (AE) architecture and adds the noise generated from the normal distribution into images Raval et al. (2017) for privacy protection. (2) Style Translator based model: it changes the style of video frames to hide visual information based on cycle-GAN architecture Wu et al. (2019a).

All the experiment results are analyzed in Subsections 2.5.2, 2.5.3, 2.5.4, and 2.5.5. In this paper, video frames are presented to illustrate the effectiveness of our F2F and V2V models. More results of video and video frames can be found in `https://github.com/ahahnut/cycle-VQ-VAE`, and you can also create your own datasets for training using our

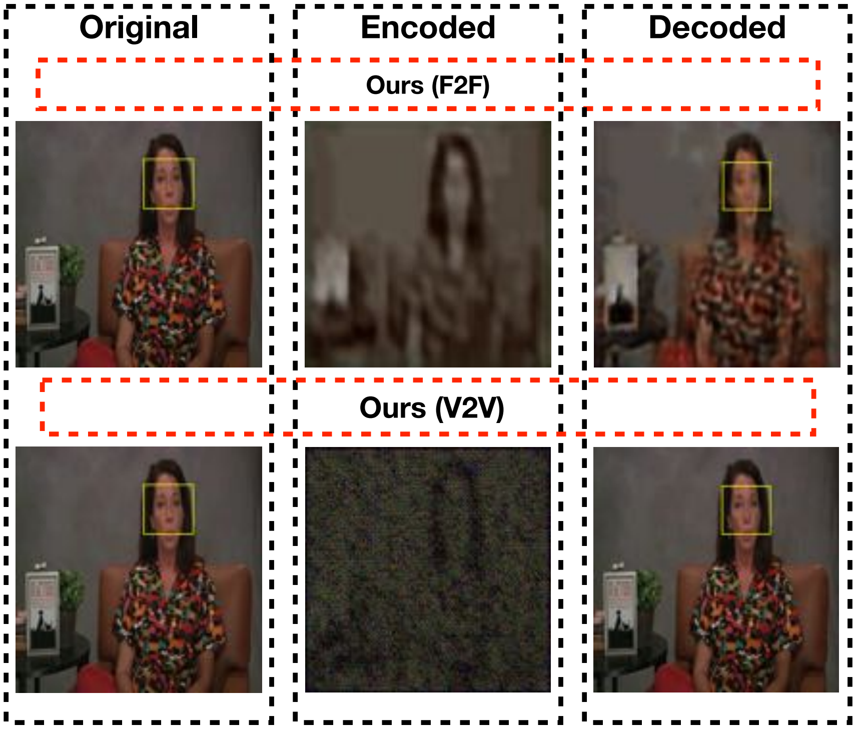

Figure 2.8 Face Detection Comparison

open-source codes.

2.5.2 Qualitative Evaluation

There are original video frames, encoded video frames, and decoded video frames in the whole process of our cycle-VQ-VAE framework.

2.5.2.1 Video Frames of F2F and V2V

We show video frames in different phases in F2F and V2V models for performance comparison. For the encoded/decoded video frames generated by F2F and V2V models, the results of face detection are presented in Fig. 2.6 and Fig. 2.7, and the results of semantic segmentation are presented in Fig. 2.9 and Fig. 2.10. Compared with the original video frames, we

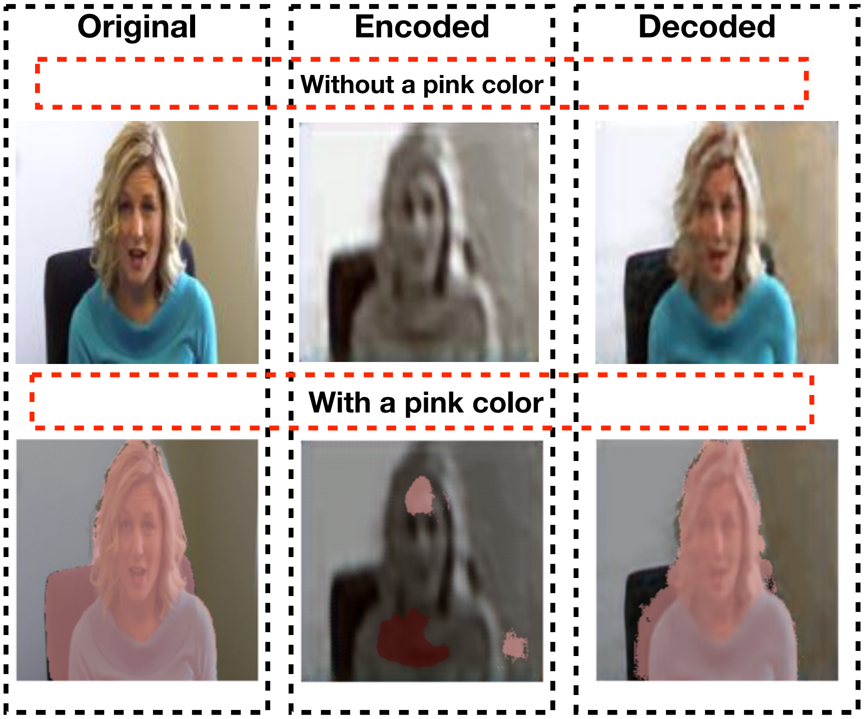

Figure 2.9 Semantic Segmentation on F2F Video Frames

can draw a conclusion that in F2F and V2V models, the encoded video frames lose sufficient visual information to resist detection while the decoded video frames can recover the lost visual information effectively for the detection task.

From Fig. 2.8 and Fig. 2.11, one can see that by utilizing the relations between frames for video processing, V2V model outperforms F2F model in terms of video compression and video reconstruction. In Fig. 2.8, the encoded video frame of V2V is harder to be recognized, and the decoded frame of V2V is clearer for face detection. In Fig. 2.11, the encoded video frame of V2V loses more visual information causing worse semantic segmentation performance, and the decoded video frame of V2V has a higher visual quality for better semantic segmentation.

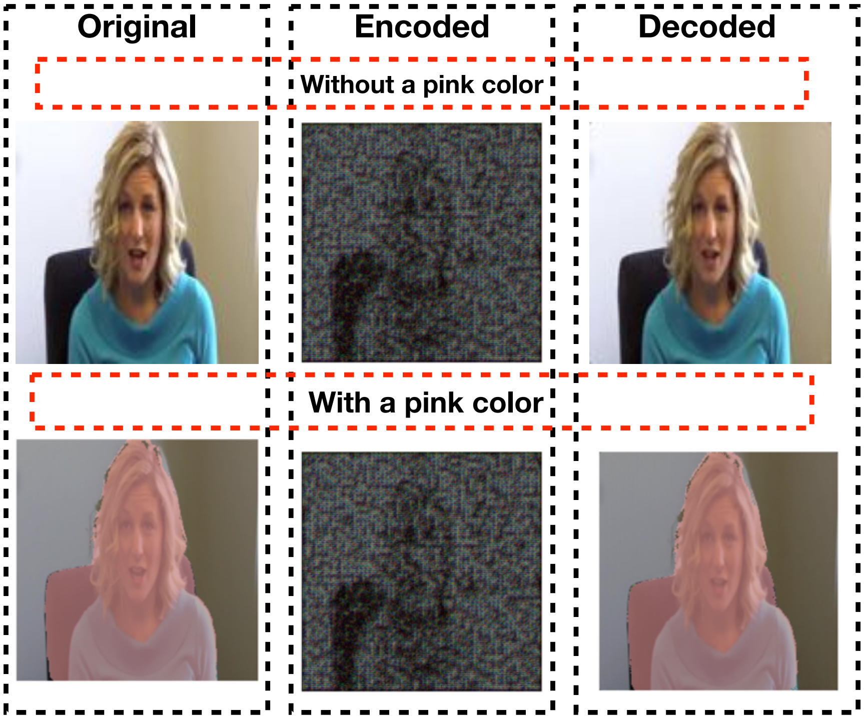

Figure 2.10 Semantic Segmentation on V2V Video Frames

2.5.2.2 Encoded Video Frames

In Fig. 2.12, the encoded video frames in F2F and V2V cannot be detected by the face detector with a rectangle, but those of the AE based model and the Style Translator based model can be detected by the face detector. From Fig. 2.14, one can see that in our F2F and V2V models, human cannot be segmented by the semantic segmentation model from the encoded video frames, but in the AE based model and the Style Translator based model, human body can be segmented correctly. The main reason why our two models perform better is that the noise (*i.e.*, the extracted audio) of F2F and V2V does not follow any patterned distribution, greatly disturbs the visual information, and reduces the detection accuracy. Besides, V2V outperforms F2F in the video compression process due to consideration of the relations

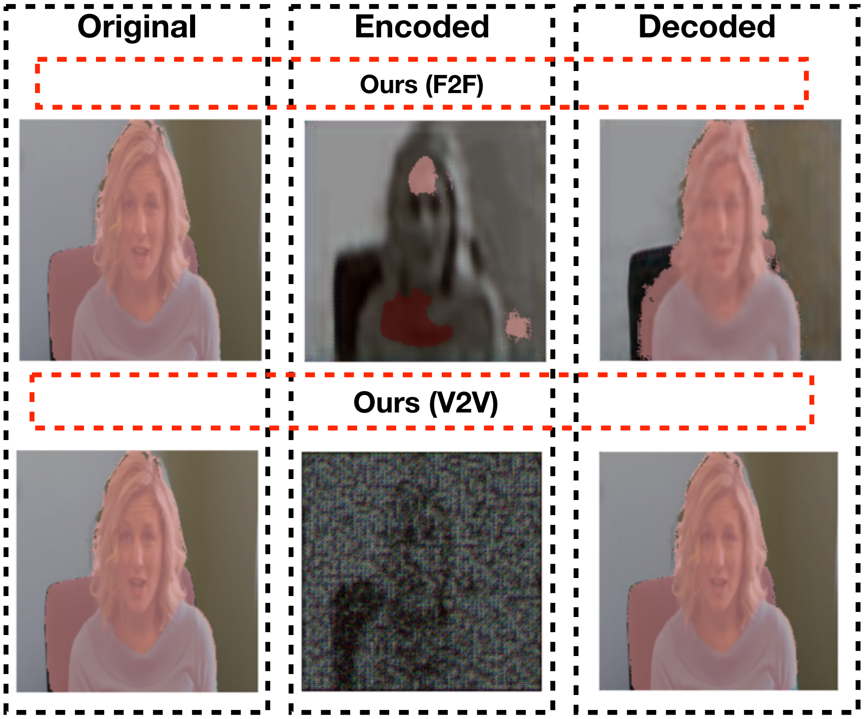

Figure 2.11 The Results of Face Detection and Semantic Segmentation in F2F and V2V

between frames even if they are both trained by our proposed cycle-VQ-VAE framework.

Moreover, since the noise can be filtered from real data by analyzing energy distribution Boyat & Joshi (2014) for performance comparison. From Fig. 2.16, we observe that the energy distribution of original frames looks like a valley. Similarly, in Fig. 2.18 and Fig. 2.19, the energy distribution of the encoded frames of the two baselines only has one valley, which indicates that it is possible to recover the original frames from the encoded ones by removing the patterned noise in real applications. Differently, in Fig. 2.17 and Fig. 2.20, the energy distribution of encoded video frames of F2F and V2V contain several valleys, which means that our extracted audio can disturb the video information in a proper low-dimensional space where the audio energy can effectively influence the energy distribution of video frames. As

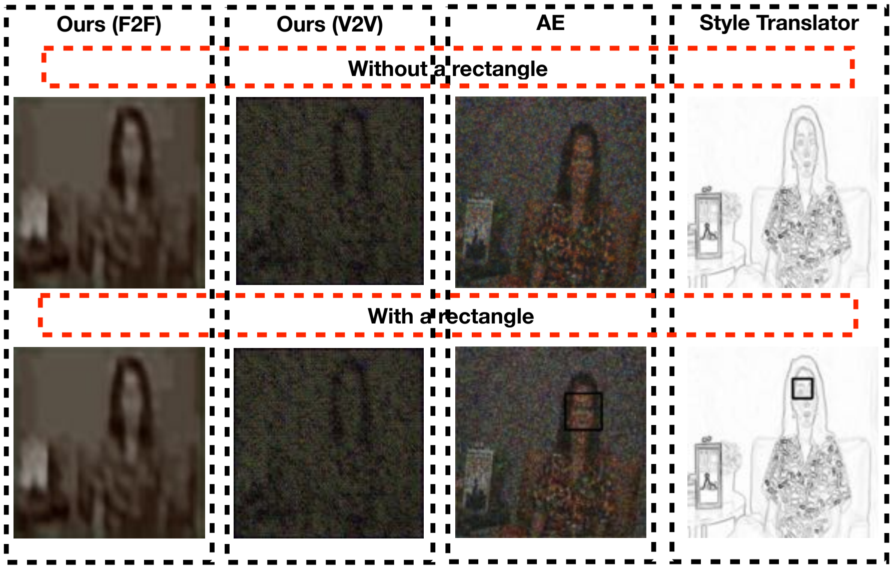

Figure 2.12 Face Detection on Encoded Video Frames: Ours *v.s.* Others

a result, it becomes harder to recover the original frames from our encoded video frames just by removing the noise. Particularly, when comparing Fig. 2.17 with Fig. 2.20, we can find out that the energy distribution of encoded video frame in V2V is more irregular than that of encoded video frame in F2F because V2V achieves a better video compression performance by taking the relations between frames into consideration, leading to a larger difficulty in removing the noise for recovery.

2.5.2.3 Decoded Video Frames

As shown in Fig. 2.13, the decoded video frames of the four models can be observed. However, only the decoded video frames of our F2F and V2V models can be detected by the face

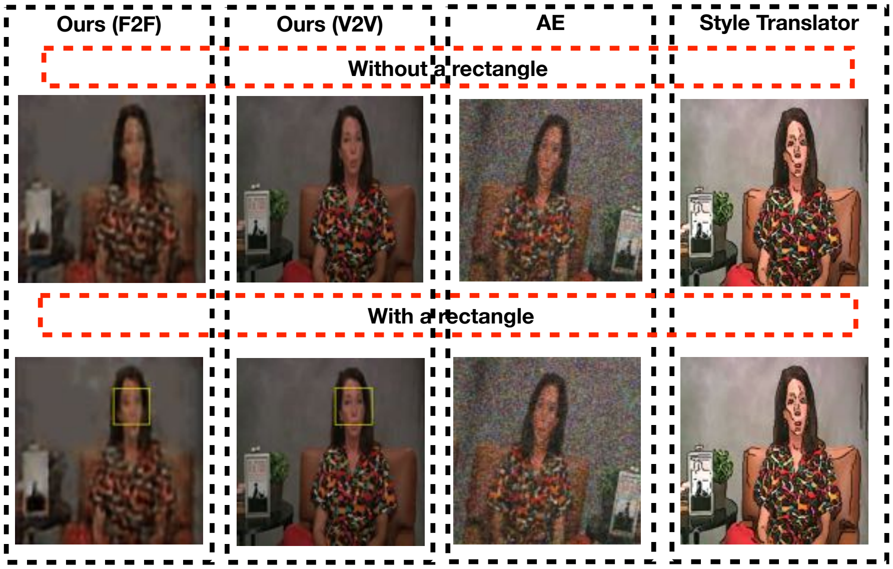

Figure 2.13 Face Detection on Decoded Video Frames: Ours *v.s.* Others

detection model with a rectangle. Similarly, in Fig. 2.15, only the decoded video frames of F2F and V2V models can be segmented with a pink color through the semantic segmentation model. It is worth mentioning that the decoded video frames should have satisfied visual quality for observation/detection in real applications. From Fig. 2.13 and Fig. 2.15, we can see that our models can make the decoded video frames maintain the expected visual quality but the two baselines fail to make it, indicating that our models outperform the two baselines. In addition, compared with the decoded video frames in F2F, the decoded video frames in V2V can be better reconstructed when considering the relations between frames with respect to the video reconstruction task.

One more same video frame is chosen to compare our models with two baselines qual-

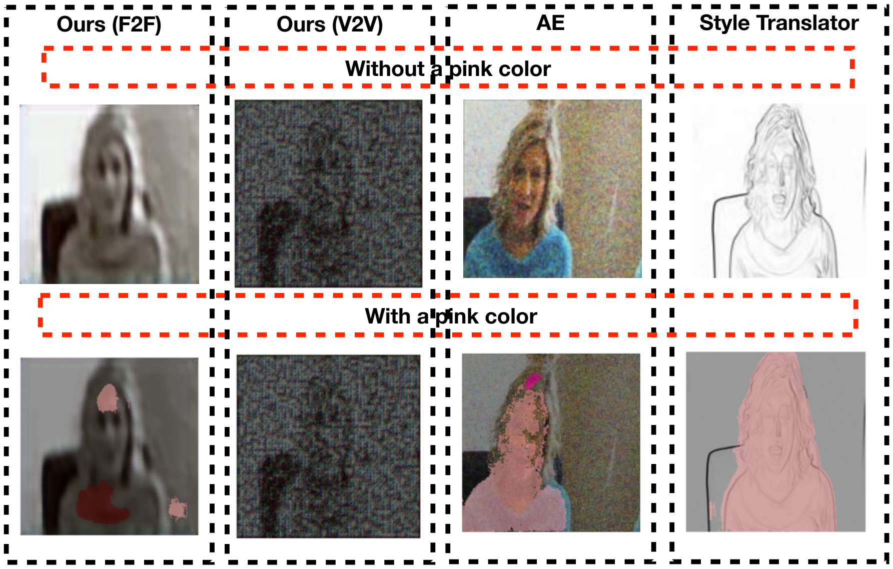

Figure 2.14 Semantic Segmentation on Encoded Video Frames: Ours *v.s.* Others

itatively for better illustrating the superiority of our models, especially V2V model. From Fig. 2.21, we observe that the encoded video frames in AE based and Style Translator based models can be detected by the face detection model, but the encoded video frames in F2F and V2V models cannot be detected, which means that our models outperform the two baselines. Especially, the encoded video frames in F2F model, AE based model, and Style Translator based model can be more or less segmented by the semantic segmentation model, but the encoded video frame in V2V model can not be segmented, indicating that V2V has the best performance of video compression and privacy protection. The results of Fig. 2.22 show that the decoded video frames in the four models can be detected by the face detection model and the semantic segmentation model, which means that F2F and V2V models can

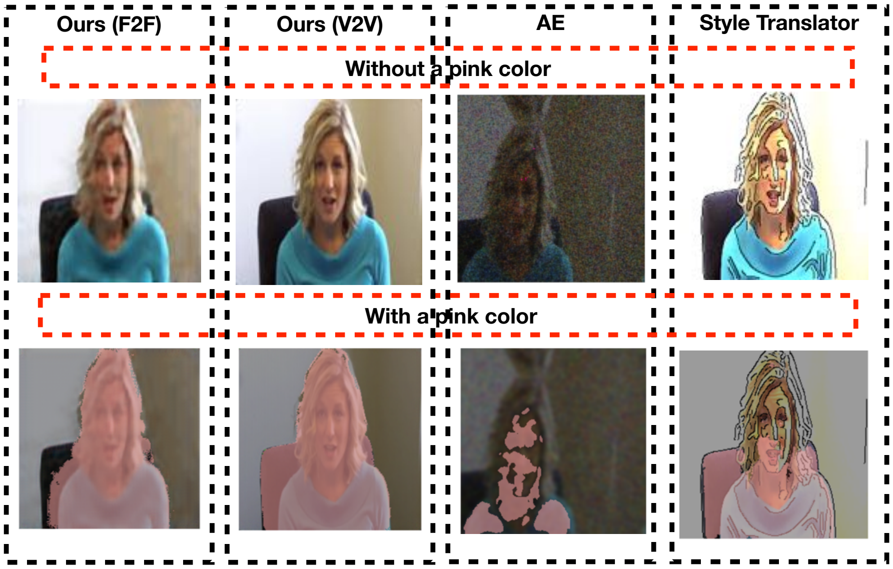

Figure 2.15 Semantic Segmentation on Dencoded Video Frames: Ours *v.s.* Others

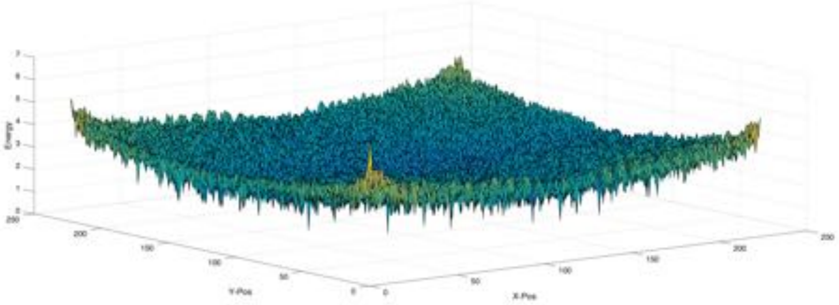

Figure 2.16 Energy Distribution of Original Video Frame (Original)

be used in video reconstruction. However, the decoded video frame in V2V has the highest visual quality, illustrating the advantage of V2V model in video reconstruction.

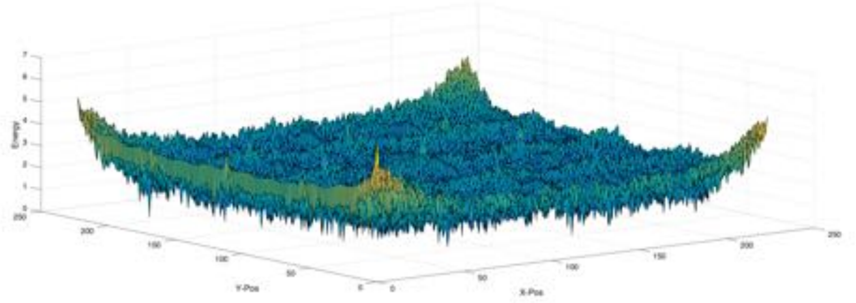

Figure 2.17 Energy Distribution of Encoded Video Frame (Ours (F2F))

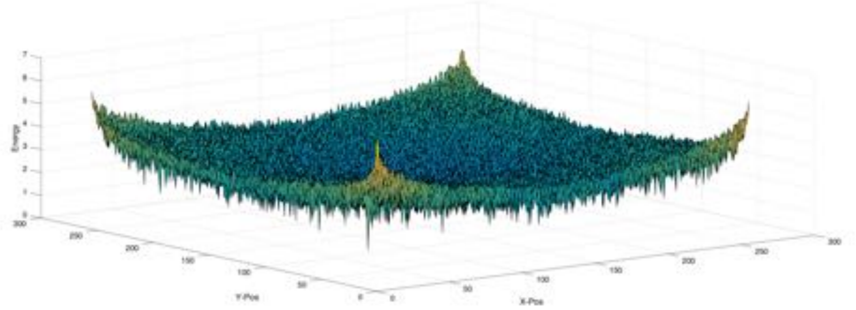

Figure 2.18 Energy Distribution of Encoded Video Frame (AE)

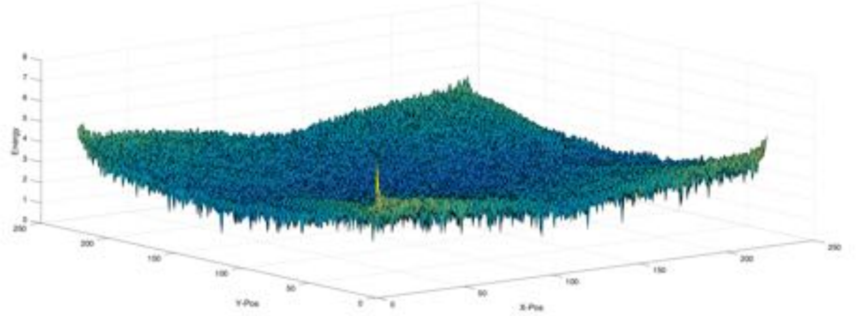

Figure 2.19 Energy Distribution of Encoded Video Frame (Style Translator)

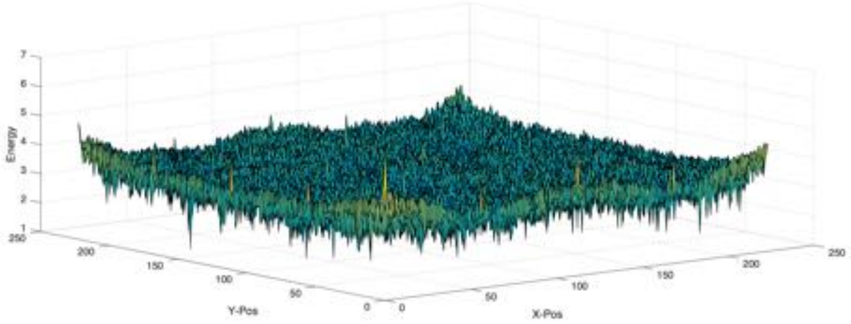

Figure 2.20 Energy Distribution of Encoded Video Frame (Ours (V2V))

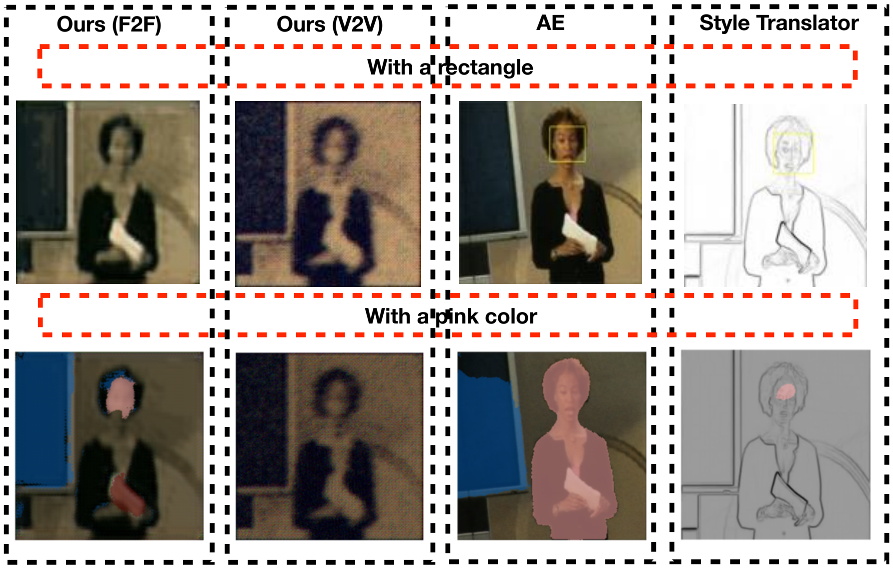

Figure 2.21 Face Detection (Top) and Semantic Segmentation (Bottom) on Encoded Frames: Ours *v.s.* Others

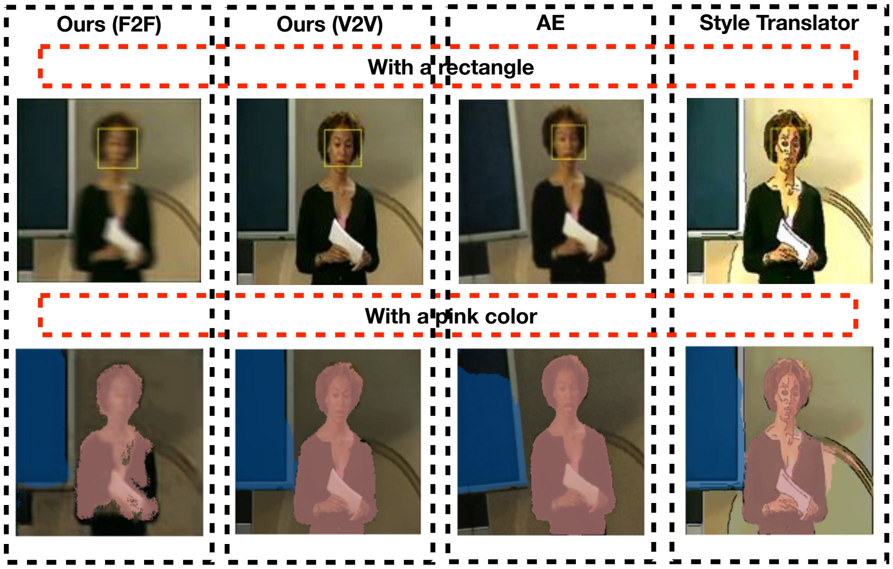

Figure 2.22 Face Detection (Top) and Semantic Segmentation (Bottom) on Dencoded Frames: Ours *v.s.* Others

2.5.3 Quantitative Evaluation

We evaluate the quantitative performance of F2F and V2V models in terms of the average accuracies of face detection and semantic segmentation, and present the results in Table 2.1 and Table 2.2.

Table 2.1 Accuracy of Face Detection

	Ours(F2F)	Ours(V2V)	AE	Style Translator
Original	96.67%	96.67%	96.67%	96.67%
Encoded	6.00%	0.00%	26.67%	36.67%
Decoded	80.00%	96.67%	46.67%	63.33%

Table 2.2 Accuracy of Semantic Segmentation

	Ours(F2F)	Ours(V2V)	AE	Style Translator
Original	93.30%	93.30%	93.30%	93.30%
Encoded	6.70%	0.00%	20.00%	36.67%
Decoded	73.33%	93.30%	43.30%	60.00%

2.5.3.1 Video Frames of F2F and V2V

Compared with the average accuracy of face detection on the original video frames (*i.e.*, 96.67% in Table 2.1), this accuracy is only 6.00% for the encoded video and can reach 80.00% for the decoded video in F2F model, and this accuracy decreases to 0.00% for the encoded video and can be recovered back to 96.67% for the decoded video in V2V model. As shown in Table 2.2, the average accuracy of semantic segmentation on original video frames is 93.30%; by using F2F model, the accuracy decreases to 6.70% on the encoded video frames and achieves 73.33% on the encoded video frames; and by using V2V model, this accuracy is only 0.00% on the encoded video and can reach 93.30% on the decoded video. These results illustrate that our F2F and V2V models can reduce the risk of privacy leakage in the encoded video frames while successfully recovering the lost visual information in the decoded video frames for real applications. In other words, our models are effective for privacy preservation in video streaming.

2.5.3.2 Encoded Video Frames

With respect to face detection on the encoded video frames, the average accuracies in our F2F model, our V2V model, the AE based model, and the Style Translator based model are 6.00%, 0.00%, 26.67%, and 36.67%, respectively (see Table 2.1). In addition, for semantic segmentation on the encoded video frames, the average accuracies in our F2F model, our V2V model, the AE based model, and the Style Translator based model reach 6.70%, 0.00%, 20.00%, and 36.67%, respectively (see Table 2.2). From the above comparison, one can see that our F2F and V2V models can lower detection accuracy on the encoded video frames in face detection and semantic segmentation and thus perform better than the two baselines in protecting visual privacy. This is because for the video, our models utilize the extracted audio that is a type of random and non-patterned distributed noise to blur the visual information while the two baselines use patterned distributed noise. What's more, V2V can obtain a lower detection accuracy than F2F in face detection and semantic segmentation on the encoded video frames since more visual information is lost in the V2V's encoding process when taking the relations between frames into account.

2.5.3.3 Decoded Video Frames

The decoded video frames are expected to recover the lost visual information as much as possible for further utilization. From Table 2.1 and Table 2.2, one can see that a higher average accuracy of face detection/semantic segmentation on the decoded video frames is achieved by our F2F and V2V models, which means our models outperform the two baselines in terms of the visual quality of decoded video frames. In addition, by comparing F2F and V2V, the decoded video frames in V2V can better be applied in face detection and semantic segmentation tasks, which means that considering the relations between frames in V2V is helpful for reconstructing a high-quality video.

2.5.4 Security Analysis

In our F2F and V2V models, we can encode the video frames with its extracted audio and decode the encoded video frames with the same audio. The encoded video frames can (i) defend against the detection attacks using face detection and semantic segmentation during the transmission process, (ii) defend against side-channel inference attack, and (iii) only be decoded with the same audio received by the authorized receivers, which is deeply analyzed as follows.

2.5.4.1 Defense against Detection Attacks

We use two mainstream detection models to validate that our encoded video frames can prevent the visual information from being accurately detected. As shown in Table 2.1 and Table 2.2, compared with the two baselines, our F2F and V2V models obtain a lower average accuracy in both face detection and semantic segmentation for the encoded video frames. The main reason lies in the method noise generation: in our encoded video frames, the noise (*i.e.*, the extracted audio) is extracted from the video so that it owns non-patterned distribution and sufficient randomness to help improve the performance of protecting visual information; while in the two baselines, the noise is generated from patterned distribution (*i.e.* normal distribution), which can be used as prior knowledge for information detection. Moreover, compared with F2F, V2V can obtain a lower accuracy and even decrease the detection accuracy to 0.00% in both face detection and semantic segmentation for the encoded video frames. This is because considering the relations between frames is effective to encode the visual information of video frames.

2.5.4.2 Defense against Side-Channel Inference Attack

The prior work Li et al. (2016a) reveals that the traffic pattern of video streaming can be used as side-channel information to infer human's activities during the transmission even if the video streaming is encrypted by TLS/SSL. Fig. 2.23 shows that the traffic pattern of original video streaming and that of the encrypted original video streaming have a pretty high similarity.

To investigate the performance of video encoding methods in resisting the side-channel inference attack, the encoded video streaming is generated using the encoded video frames. The traffic size of the original video streaming, the encoded video streaming of F2F, the encoded video streaming of V2V, and the encoded video steaming of two baselines are presented in Fig. 2.24. Then, we use the side-channel inference method in Li et al. (2016a) to calculate the accuracy of activity inference in video streaming and report the results in Table 2.3, where the average accuracy of activity inference is 95.8% in the original video streaming. The average accuracy of activity inference is 95.60% in AE encoded video streaming and 94.50% in Style Translator encoded video streaming, indicating that these two encoding methods cannot prevent side-channel information leakage. Notably, the average accuracy of activity inference is reduced to 42.86% in the encoded video streaming of F2F and even reduced to 0.00% in the encoded video streaming of V2V. The reason is that the encoding process of our F2F and V2V model can effectively smooth the traffic pattern. In particular, the relations between frames are exploited for video compression in V2V, further increasing the difficulty of traffic analysis during transmission. Thus, we can conclude that our F2F and V2V models can effectively resist side-channel inference attack.

Moreover, experiments are conducted to compare our F2F and V2V models with two baseline models after using TLS/SSL (AES 128 bit) encryption method for video transmission, traffic size are shown in Fig. 2.25, and results of activity inference are presented in Table 2.3. In Fig. 2.25, the traffic pattern of video streaming seems almost unchanged after video encryption. In Table 2.3, the average accuracy of activity inference is 94.80% in AE encrypted encoded video streaming, 93.70% in Style Translator encrypted encoded video streaming, 41.98% in F2F encrypted encoded video streaming, and still 0.00% in V2V encrypted encoded video streaming. These results indicate that the encoding methods of AE and Style Translator cannot prevent the side-channel attack even if the encryption method is used during video transmission. On the contrary, our encoding models outperform these two baselines and can prevent the side-channel attack effectively.

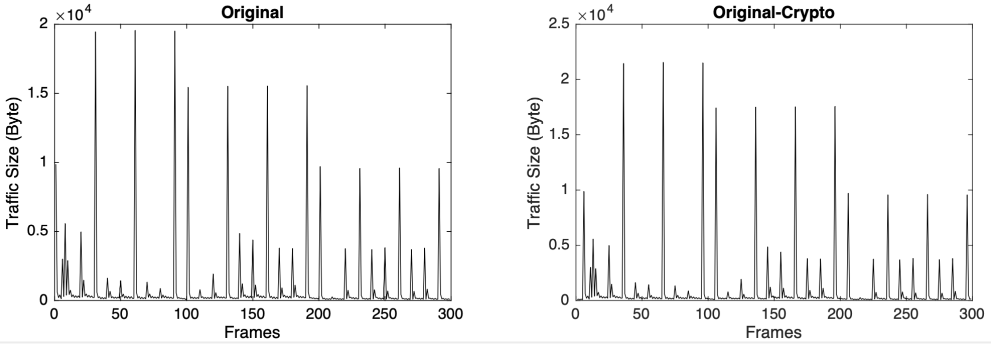

Figure 2.23 Traffic Size of Original Video Streaming before and after Encryption

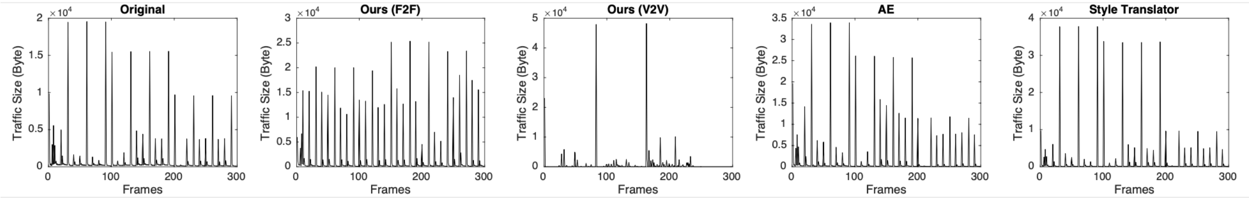

Figure 2.24 Traffic Size of Video Streaming before Encryption: Ours *v.s.* Others

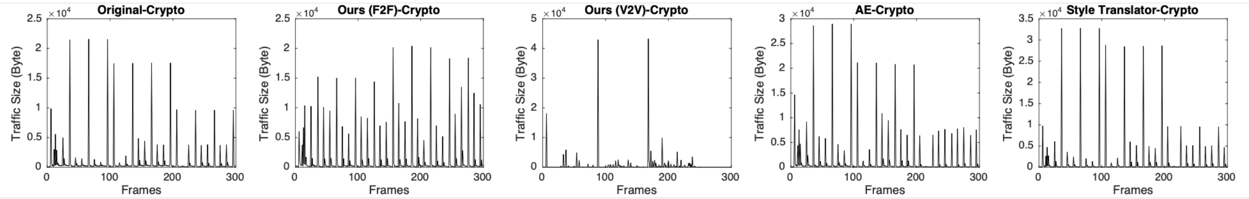

Figure 2.25 Traffic Size of Video Streaming after Encryption: Ours *v.s.* Others

2.5.4.3 Defense against Un-authorization

In our F2F and V2V models, we train the same audio to encode the video frames and decode the encoded video frames. Different from the noise that follows certain distributions (*e.g.*, normal distribution), the audio extracted from its corresponding video is unique and cannot be easily generated or manipulated. Therefore, the video streaming can only be recovered by the authorized receivers who have the extracted audio.

Table 2.3 Results of Activity Inference

	Accuracy		Accuracy
Original	95.80%	Original-Crypto	94.90%
Ours (F2F)	42.86%	Ours (F2F)-Crypto	41.98%
Ours (V2V)	0.00%	Ours (V2V)-Crypto	0.00%
AE	95.60%	AE-Crypto	94.80%
Style Translator	94.50%	Style Translator-Crypto	93.70%

2.5.5 Transmission Efficiency Analysis

Notice that the efficiency of video transmission has not yet been incorporated into visual privacy protection by the existing works, but the consideration of transmission efficiency is a necessary component for IoT devices and applications. One major advantage of our cycle-VQ-VAE framework over the state-of-the-art is that it can achieve effective visual privacy protection and efficient video transmission simultaneously. The main reason is that the encoder component in our cycle-VQ-VAE framework leverages the extracted audio to encode the corresponding video, in which the video actually is compressed to a reduced size, and the transmission time can be reduced as well. On the contrary, the previous visual privacy-preserving models (such as AE based and Style Translator based model) exploit the noise to hide the original visual content, where the additional noise increases the video size, and the transmission time is increased. Furthermore, we do real-data experiments and use the transmission time as a performance metric to illustrate the transmission efficiency of our models during video streaming transmission in real applications. In Table 2.4, we list the transmission time of uploading 10-second video streaming to an edge server at different network bandwidths. Compared with the original video, the transmission time is averagely decreased by 16.2% in our F2F model due to the video compression in the encoding process. Even better, the transmission time is averagely reduced by 53.4% in our V2V model as a better video performance can be achieved by considering the relations between frames. But the transmission time is averagely increased by 43.8% in AE based model and 9.1% in Style Translator based model, in which noise is added to disturb the original visual information without compression.

Table 2.4 Transmission Time at Different Bandwidths (Ours (F2F) v.s. Others)

	Original	Ours (F2F)	Ours (V2V)	AE	Style Translator
0.5MB/s	3.84s	3.24s(↓ 15.6%)	1.75s(↓ 54.4%)	5.6s(↑ 45.8%)	4.2s(↑ 9.3%)
1MB/s	1.87s	1.57s(↓ 16.1%)	0.87s(↓ 53.1%)	2.68s(↑ 43.3%)	2.05s(↑ 9.6%)
2MB/s	0.94s	0.78s(↓ 17.1%)	0.44s(↓ 52.7%)	1.34s(↑ 42.5%)	1.02s(↑ 8.5%)
Average		↓ 16.2%	↓ 53.4%	↑ 43.8%	↑ 9.1%

2.6 Conclusion

In this paper, we propose an audio-visual autoencoder framework, named cycle-VQ-VAE. To the best of our knowledge, this is the first work to use multi-source information to generate privacy-preserving video streaming; especially, the audio is extracted from its corresponding video and used as the random noise to disturb the visual information. Since the extracted audio is unique and meaningful, it cannot be generated or manipulated easily and thus can be used by the authorized receivers to decode the encoded video. In addition, we develop F2F and V2V models under cycle-VQ-VAE framework. The entire encoded video streaming of our models has a more smooth traffic pattern, which can prevent the side-channel inference attacks using traffic size analysis. Besides, with video compression in our encoding process, the time of video transmission can be greatly decreased. Via extensive experiments, we demonstrate that our F2F model can preserve the expected visual quality, reduce the risk of visual privacy leakage, and improve the efficiency of video transmission; especially, V2V model outperforms F2F model in all evaluation metrics owing to the consideration of the relations between frames for video compression and reconstruction.

CHAPTER 3

PRIVACY-PRESERVING MECHANISMS FOR MULTI-LABEL IMAGE RECOGNITION

3.1 Motivation

Multi-label image recognition is a fundamental component in computer vision applications Chen et al. (2019b), such as medical diagnosis recognition Ge et al. (2018b), human attribute recognition Li et al. (2016b), and retail checkout recognition George & Floerkemeier (2014); Wei et al. (2019). With the rapid development of deep neural networks, the performance of multi-label image recognition is remarkably improved via deep learning models. However, due to the reliance on massive images uploaded to third-party platforms to accomplish multi-label image recognition, these deep learning models may face a serious risk of privacy leakage Brkic et al. (2017); Cai et al. (2023); Xu et al. (2023a). For example, attackers can infer private information via extracted features and/or victim model's weights, causing substantial economic losses for individuals and institutions. More problematically, they even can launch attack mechanisms in black-box applications (APIs) by only utilizing the distribution of model's outputs Rahman et al. (2018); Truex et al. (2018). As multi-label image recognition plays a pivotally important role in many real applications, it becomes essential to guarantee privacy protection while maintaining prediction performance for the multi-label image recognition models.

Recently, researchers have realized the importance of privacy protection when designing deep neural networks in real applications He et al. (2017). One vein of research is to hide sensitive visual information by integrating noise with images for data publishing to protect privacy Raval et al. (2017); Uittenbogaard et al. (2019); Chen et al. (2018); Cai et al. (2021b); Xiong et al. (2021b); Xu et al. (2021). However, due to the lack of theoretical privacy guarantee, the performance of those methods heavily rely on discriminator. On the other hand, differential privacy mechanisms are adopted in many deep learning models Zheng et al. (2020a); Zhu & Philip (2019); Wu et al. (2020) to theoretically achieve privacy guarantee for different goals, such as generation De et al. (2022) and determination. In these deep learning models, noise is usually employed to disturb the models' weights in order to keep the models' parameters secure Wu et al. (2017); McMahan et al. (2018); Xia et al. (2019); Xu et al. (2019); Xiong et al. (2023b), or integrated into the models' input features so as to generate privacy-preserving data for public publishing Phan et al. (2017); Hitaj et al. (2017); Xiong et al. (2021a). However, large aggregated noise brought by deep structure will result in low performance and poor model usability in real applications. Moreover, black-box attack, which can be easily implemented only using the model's outputs, is not considered in the existing works. The aforementioned observations motivated us to work out a solution to ensure privacy protection, maintain prediction accuracy, alleviate the aggregated noise's side effect, and defend black-box attack simultaneously for the multi-label image recognition models.

In this paper, we propose P2-ML-GCN mechanism that satisfies ϵ-differential privacy on the outputs of Multi-label Graph Convolutional Networks (ML-GCN) Chen et al. (2019b) with the intention of preventing black-box attack. To further increase the prediction accuracy of P2-ML-GCN, we develop RP2-ML-GCN, where a regularization term is designed to enhance the model's robustness, and the global sensitivity in differential privacy mechanism is smoothed via a proper bound to mitigate excessive noise's side effect. In other words, we can enhance the prediction accuracy using a regularization term and/or a bounded global sensitivity, which pioneers a new research direction for effectively designing privacy-preserving deep learning algorithms. Moreover, through rigorous theoretical analysis, we prove the guarantee of privacy protection for ML-GCN, the effectiveness of our proposed regularization term for robustness improvement, the advantage of utilizing a bounded global sensitivity to alleviate excessive noise's side effect, and the capability of our proposed models to protect the privacy of model's weights and input features. Finally, we evaluate the performance of our proposed models by conducting intensive real-data experiments and comparing them with the-state-of-the-art models. Our multifold contributions are addressed as follows.

- To the best of our knowledge, this is the first work to design privacy-preserving multi-label image recognition models based on differential privacy mechanism.

- Our first model P2-ML-GCN applies differential privacy mechanism on the model's outputs, which can defend black-box attack and avoid large aggregated noise even if a neural network has many layers.

- To improve the prediction accuracy of P2-ML-GCN, a regularization term is designed in our second model RP2-ML-GCN to enhance the model's robustness, and a proper bound of global sensitivity in differential privacy mechanism is set to alleviate the side effect of excessive noise.

- Through rigorous theoretical analysis, we prove that our two proposed models are able to protect the privacy of the model's outputs, weights and input features with the guarantee of ϵ-differential privacy, which provides a guidance for the design of privacy-preserving deep learning algorithms.

- Comprehensive experiments are well-conducted to validate the advantages of P2-ML-GCN and RP2-ML-GCN.

The rest of this paper is organized as follows. Related works are briefly summarized in Section 3.2. After introducing preliminaries in Section 3.3, we detail our models in Section 3.4. In Section 3.5, we conduct real-data experiments and analyze all results. Finally, we end up with a conclusion in Section 3.6.

3.2 Related Works

The state-of-the-art about multi-label image recognition and differential privacy-based machine learning algorithms is summarized in the following.

3.2.1 Multi-label Image Recognition

A straightforward idea of multi-label recognition is to train independent binary classifiers for each object label based on state-of-the-art deep Convolutional Neural Networks (CNNs) Huang et al. (2017); Simonyan & Zisserman (2014); Szegedy et al. (2016), which, however, ignores the relationship among labels. To improve the efficiency of multi-label image recognition models, the label correlation is taken into account in some works Wang et al.

(2016a); Zhu et al. (2017); Wang et al. (2017); Chen et al. (2019b). Wang *et al.* considered the correlation of labels through employing Recurrent Neural Networks (RNNs) in embedded label vectors Wang et al. (2016a). Zhu *et al.* studied both semantic and spatial relations of multiple labels to design a spatial regularization network based on weighted attention maps Zhu et al. (2017). Wang *et al.* proposed a spatial transformer layer and Long-Short Term Memory (LSTM) units to capture label correlation Wang et al. (2017). Recently, Chen *et al.* proposed a GCN-based Multi-label Graph Convolutional Networks (ML-GCN) model, which applies the directed graph of multiple object labels built by labels' co-occurrence pattern in dataset Chen et al. (2019b). So far, the method of Chen et al. (2019b) outperforms other existing methods. However, the study about how to design a privacy-preserving model for such multi-label image recognition has been overlooked by the existing works.

3.2.2 Differential Privacy in Deep Learning

Differential privacy mechanism was proposed by Dwork *et al.* for privacy guarantee on adjacent databases Dwork et al. (2006). The incorporation of differential privacy mechanisms and deep learning algorithms in most of the existing works can be briefly divided into two categories. One is to update weights in stochastic gradient descent (SGD) algorithms with additional noise calculated by the gradient bound Abadi et al. (2016); Wu et al. (2017); McMahan et al. (2018); Xia et al. (2019); Xu et al. (2019); He et al. (2023); Wang et al. (2023b), or to update weights in regression models with additional noise calculated by the polynomial coefficient of the regression models' parameters Phan et al. (2016), which mainly focuses on the parameters of learning models to satisfy differential privacy requirements. The other is to obtain a privacy-preserving generative model by employing a proper noise, which keeps an eye on input features Yoon et al. (2019); Phan et al. (2017); Hitaj et al. (2017); Beaulieu-Jones et al. (2019). But, when the number of input features and the number of shared parameters are large, these existing works sacrifice a high privacy budget to maintain models' accuracy. In addition, since differential privacy mechanisms are implemented on either weights or features in every layer of deep learning models, these existing works may suffer from a large aggregated noise when a neural network contains too many layers. Moreover, even if these works can obtain secure weights and features, they cannot resist black-box attack that can be accomplished based on the distribution of models' outputs Shokri et al. (2017); Rahman et al. (2018).

In this paper, in order to defend black-box attack and protect privacy for multi-label image recognition, we propose two novel models, including P2-ML-GCN and RP2-ML-GCN, by implementing differential privacy mechanisms on ML-GCN's outputs. Compared with the state-of-the-art, our models have three major advantages: i) the noise added into outputs can be bounded even if the neural network has many layers, which can significantly reduce the aggregated noise of an entire model and thus provide a higher degree of privacy guarantee; ii) the two proposed models can prevent the aforementioned black-box attack because the noise disturbs the distribution of outputs; and iii) in RP2-ML-GCN, a regularization item based on the Frobenius norm of weights of classifiers is added to the loss function for the performance improvement, and a bound of global sensitivity in differential privacy mechanisms is set appropriately to mitigate the excessive noise's side effect in P2-ML-GCN. Finally, we rigorously prove that our proposed mechanisms can provide a helpful guidance for the design of privacy-preserving deep learning algorithms.

3.3 Preliminaries

In this section, we introduce graph convolutional network (GCN), ML-GCN model for multi-label image recognition Chen et al. (2019b), and the basics of differential privacy Dwork et al. (2006).

3.3.1 Graph Convolutional Network

Graph Convolutional Network (GCN) was introduced in Kipf & Welling (2017) to perform semi-supervised graph classification aiming to update the node representations of a graph by convolutional operations. The two inputs of GCN include the node feature matrix in the l-th layer $H^l \in \mathbb{R}^{n \times d}$ and the node correlation matrix $A \in \mathbb{R}^{n \times n}$, where n denotes the number of nodes in a graph and d is the dimension of node features in l-th layer. After employing the convolutional operations of Kipf & Welling (2017), the node feature matrix $H^{l+1} \in \mathbb{R}^{n \times d'}$ in the $(l+1)$-th layer can be represented as $H^{l+1} = h(\hat{A}H^lW^l)$, where $h(\cdot)$ denotes a non-linear operation, $\hat{A} \in \mathbb{R}^{n \times n}$ is the normalized version of correlation matrix A, and $W^l \in \mathbb{R}^{d \times d'}$ is a transformation matrix to be learned.

3.3.2 ML-GCN

By taking the label correlation into account, ML-GCN outperforms other existing approaches in multi-label image recognition Chen et al. (2019b) and thus is adopted as our baseline.

In Chen et al. (2019b), a directed graph is built on all images of a dataset, where the vertices represent object labels, and the weight of a directed edge is the occurrence probability of head vertex when its corresponding tail vertex occurs. The directed graph is used to mine co-occurrence patterns of object labels within the dataset through Graph Convolutional Networks (GCN). The image features can be extracted by Resnet-101 He et al. (2016). Then, the co-occurrence pattern can be combined with features to improve the performance of multi-labels recognition.

Let C be the number of labels' categories, D be the dimension of features, and $\hat{y} \in \mathbb{R}^C$ be the output prediction labels. We can obtain $\hat{y}$ via Eq. (3.1).

$$\hat{y} = Wx, \tag{3.1}$$

where $W \in \mathbb{R}^{C \times D}$ is the final parameter matrix after GCN has been trained, and $x \in \mathbb{R}^D$ is the feature vector extracted by Resnet-101.

Finally, ML-GCN is trained with the following multi-label classification loss function.

$$L = \sum_{i=1}^{C} y_i \log(\sigma(\hat{y}_i)) + (1 - y_i) \log(1 - \sigma(\hat{y}_i)), \tag{3.2}$$

where $y_i \in \{0, 1\}$ is the real label of i-th category, $\hat{y}_i \in [0, 1]$ is the confidence score of i-th category, and $\sigma(\cdot)$ is the sigmoid function Yin et al. (2003).

3.3.3 Differential Privacy

Differential privacy defines a mathematical measurement of data privacy protection for a dataset Dwork et al. (2006).

Definition 1. *A randomized mechanism, $\mathcal{M}$ ($U \rightarrow \mathbb{R}$), satisfies ϵ-differential privacy, if for any two adjacent inputs $u, u' \in U$ and any $S \subset \mathbb{R}$, there is*

$$\Pr[\mathcal{M}(u) \in S] \leq e^{\epsilon} \Pr\left[\mathcal{M}\left(u'\right) \in S\right], \tag{3.3}$$

where ϵ is a positive real number and quantifies information leakage.

To achieve ϵ-differential privacy, $\mathcal{M}$ can be constructed by a Laplace mechanism based on any real-value function f.

With respect to f, the global sensitivity S_f is defined as the maximum absolute distance between any two adjacent inputs in U Soria-Comas et al. (2017); Lundmark & Dahlman (2017); Kasiviswanathan et al. (2011), i.e.,

$$S_f = \sup_{u,u' \in U} |f(u) - f(u')|_1. \tag{3.4}$$

The randomized mechanism, $\mathcal{M}$, which satisfies ϵ-differential privacy for function f, can be obtained via additive Laplace noise as follows.

$$\mathcal{M}(u) = f(u) + Lap\left(0, S_f/\epsilon\right), \tag{3.5}$$

in which $Lap(0, S_f/\epsilon)$ is the Laplace distribution.

3.4 Methodology

In this section, we elaborate on the details of our proposed models, including P2-ML-GCN and RP2-ML-GCN. In P2-ML-GCN, to achieve privacy-preserving multi-label image recognition, we apply differential privacy mechanism to ML-GCN's prediction outputs based on additive Laplace noise. Notice that the prediction accuracy of P2-ML-GCN may be reduced due to the added additive Laplace noise. Hence, to further improve the image recognition performance, we propose RP2-ML-GCN that enhances the model's robustness with the help of a regularization term. Moreover, we analyze the relationship of privacy guarantee between our proposed models that implement differential privacy mechanisms on the prediction outputs and the approaches that adopt differential privacy mechanisms on input features or parameters, which confirms the effectiveness of our proposed models. Finally, we extend our findings to a more general case to offer a guidance for the design of privacy-preserving deep learning approaches. Since it is hard to show all analysis of multi-layer neural network with limited page length, in this paper we mainly focus on analyzing the bias of loss function and the performance of differential privacy for model weights and features in a single layer perceptron.

3.4.1 Privacy-Preserving ML-GCN

In P2-ML-GCN, we implement differential privacy mechanism on ML-GCN's prediction output vector, $\hat{y}$, in order to make the model's outputs satisfy ϵ-differential privacy, in which Laplace noise is utilized to disturb ML-GCN's outputs instead of its input features or parameters to resist black-box attack. According to Laplace mechanism, there are two steps to establish a randomized mechanism satisfying ϵ-differential privacy. First, we denote the global sensitivity of $\hat{y}$ as $S_{\hat{y}}$. Second, from Eq. (3.5), we can obtain a randomized mechanism $\hat{y}'$ that satisfies ϵ-differential privacy by adding the Laplace noise $Lap(0, \frac{S_{\hat{y}}}{\epsilon})$ to the output vector $\hat{y}$ as shown in Eq. (3.6), where $\hat{y} \in \mathbb{R}^C$ and α generated from $Lap(0, \frac{S_{\hat{y}}}{\epsilon})$ are

C-dimension vectors.

$$\hat{y}' = \hat{y} + \alpha. \tag{3.6}$$

Theorem 1. *Given the Laplace noise $Lap(0, \frac{S_{\hat{y}}}{\epsilon})$ added into the output vector $\hat{y}$, each element $\hat{y_i}'$ in the disturbed output vector $\hat{y}'$ satisfies ϵ-differential privacy.*

Proof. Let $\Pr[\cdot]$ be a commonly designed Laplace distribution Eltoft et al. (2006). Accordingly, we have,

$$\ln \frac{\Pr[\hat{y_i}]}{\Pr[\hat{y_i}']} = \ln \frac{\frac{\epsilon}{2S_{\hat{y_i}}} e^{-\frac{\epsilon}{S_{\hat{y_i}}}|\hat{y_i}|}}{\frac{\epsilon}{2S_{\hat{y_i}}} e^{-\frac{\epsilon}{S_{\hat{y_i}}}|\hat{y_i}'|}} = \frac{\epsilon}{S_{\hat{y_i}}}(|\hat{y_i}'| - |\hat{y_i}|) \leq \epsilon. \tag{3.7}$$

Eq. (3.7) shows that each element $\hat{y_i}'$ in the disturbed output vector $\hat{y}'$ satisfies ϵ-differential privacy. □

Theorem 1 demonstrates that our proposed model P2-ML-GCN can provide the multi-label image recognition with differential privacy guarantee. Correspondingly, the loss function of multi-label image recognition in P2-ML-GCN can be expressed by the disturbed output vector $\hat{y}'$ in Eq. (3.8).

$$\begin{aligned} L_{P2} =& \sum_{i=1}^{C} y_i \log(\sigma(\hat{y}')) + (1 - y_i) \log(1 - \sigma(\hat{y}')) \\ =& \sum_{i=1}^{C} y_i \log(\sigma(\hat{y_i} + Lap(0, \frac{S_{\hat{y}}}{\epsilon}))) \\ &+ (1 - y_i) \log(1 - \sigma(\hat{y_i} + Lap(0, \frac{S_{\hat{y}}}{\epsilon}))). \end{aligned} \tag{3.8}$$

During the training process of P2-ML-GCN, we intend to minimize L_{P2} to improve the prediction accuracy of ML-GCN while ensuring ϵ-differential privacy. In addition, we can control the privacy protection degree by adjusting the value of ϵ. Particularly, a smaller ϵ indicates a higher privacy protection degree.

3.4.2 Robust Privacy-Preserving ML-GCN

The noise added in P2-ML-GCN indeed offers differential privacy guarantee, but may also reduce the prediction accuracy of ML-GCN. Therefore, we design a more robust model, RP2-ML-GCN, to alleviate the influence on the prediction accuracy of multi-label image recognition while gaining the same degree of differential privacy guarantee. Specifically, in RP2-ML-GCN, we integrate the loss function of P2-ML-GCN with a regularization term to increase the prediction accuracy of ML-GCN.

There are three phases in RP2-ML-GCN. i) In the first phase, we simplify the traditional multi-label loss function for better theoretical analysis. ii) In the second phase, we calculate the bias of loss function to analyze the influence of the additive Laplace noise on the prediction accuracy of multi-label image recognition model. iii) In the third phase, we theoretically prove that the regularization term can improve the model's robustness from the viewpoint of linear regression.

3.4.2.1 Function Simplification

Since the sigmoid function is differentiable at the point 0, we can obtain an approximate quadratic polynomial in Eq. (3.9) through Taylor Theorem Rababah (1993) at the point 0.

$$log(1 + e^{-\hat{y}_i}) \approx \log 2 - \frac{1}{2}\hat{y}_i + \frac{1}{8}(\hat{y}_i)^2. \tag{3.9}$$

Then, we simplify the traditional multi-label loss function via the sigmoid function and its approximate quadratic polynomial function. The simplification process of traditional multi-label loss function is presented as follows:

$$\begin{aligned}
L =& \sum_{i=1}^{C} y_i \log(\sigma(\hat{y_i})) + (1 - y_i)\log(1 - \sigma(\hat{y_i})) \\
=& \sum_{i=1}^{C} -y_i \log(1 + e^{-\hat{y_i}}) + (1 - y_i)(\log(e^{-\hat{y_i}})) \\
& + (y_i - 1)\log(1 + e^{-\hat{y_i}}) \\
=& \sum_{i=1}^{C} y_i\hat{y_i} - \hat{y_i} - \log(1 + e^{-\hat{y_i}}) \\
\approx& \sum_{i=1}^{C} y_i\hat{y_i} - \hat{y_i} - (\log 2 - \frac{1}{2}\hat{y_i} + \frac{1}{8}(\hat{y_i})^2) \\
=& \sum_{i=1}^{C} -\frac{1}{8}(\hat{y_i})^2 - \frac{1}{2}\hat{y_i} + y_i\hat{y_i} - \log 2.
\end{aligned} \tag{3.10}$$

By substituting Eq. (3.1) into Eq. (3.10), we obtain Eq. (3.11).

$$\begin{aligned}
L =& \sum_{i=1}^{C} -\frac{1}{8}(\hat{y_i})^2 - \frac{1}{2}\hat{y_i} + y_i\hat{y_i} - \log 2 \\
=& -\frac{1}{8}(Wx)^T(Wx) - (\frac{1}{2} - y_i)(Wx) - C\log 2,
\end{aligned} \tag{3.11}$$

where W is the parameter matrix learned by GCN, x is feature vector extracted by Resnet-101, $y_i \in \{0, 1\}$ is the groundtruth label of i-th category, and C is the number of categories.

3.4.2.2 Bias Analysis

According to Eq. (3.11), we can rewrite the loss function of P2-ML-GCN in Eq. (3.12).

$$\begin{aligned}
L_\alpha =& -\frac{1}{8}((Wx + \alpha)^T(Wx + \alpha)) \\
& - (\frac{1}{2} - y_i)(Wx + \alpha) - C\log 2.
\end{aligned} \tag{3.12}$$

In the analysis of machine learning algorithms, the bias of loss function is typically used to investigate the influence of the additive noise on the prediction accuracy.

Lemma 1. *The expectation of Laplace noise* $Lap(0, \frac{S_{\hat{y}}}{\epsilon})$ *is*

$$\mathbb{E}(Lap(0, \frac{S_{\hat{y}}}{\epsilon})) = 0.$$

Lemma 2. *The expectation of square Laplace noise* $\mathbb{E}(Lap(0, \frac{S_{\hat{y}}}{\epsilon})^2)$ *is equal to*

$$\mathbb{E}(Lap(0, \frac{S_{\hat{y}}}{\epsilon})) + \mathrm{Var}(Lap(0, \frac{S_{\hat{y}}}{\epsilon})) = \frac{2S_{\hat{y}}}{\epsilon^2}.$$

According to Lemma 1, Lemma 2, Eq. (3.11) and Eq. (3.12), the bias of loss function, denoted by $\mathbb{E}(\Delta L)$, can be calculated via Eq. (3.13).

$$\begin{aligned}
\mathbb{E}(\Delta L) &= \mathbb{E}(|L_\alpha - L|) \\
&= \mathbb{E}(| -\frac{1}{8}\alpha^T W x - \frac{1}{8}x^T W^T \alpha - \frac{1}{8}\alpha^T \alpha - (\frac{1}{2} - y_i)\alpha|) \\
&= | -\frac{1}{8}\mathbb{E}(\alpha^T)\mathbb{E}(Wx) - \frac{1}{8}\mathbb{E}(x^T W^T)\mathbb{E}(\alpha) - \frac{1}{8}\mathbb{E}(\alpha^T \alpha) - (\frac{1}{2} - y_i)\mathbb{E}(\alpha)| \\
&= | -\frac{1}{8}\mathbb{E}(Lap(0, \frac{S_{\hat{y}}}{\epsilon})^2) \qquad (3.13) \\
&= | -\frac{1}{8} \times \frac{2S_{\hat{y}}}{\epsilon^2}| \\
&= | -\frac{S_{\hat{y}}}{4\epsilon^2}| \\
&= \frac{S_{\hat{y}}}{4\epsilon^2}.
\end{aligned}$$

From the expression of $\mathbb{E}(\Delta L)$, we can see that there exists an inverse proportion between $\mathbb{E}(\Delta L)$ and ϵ; that is, the smaller ϵ is, the greater $\mathbb{E}(\Delta L)$ is. In other words, a higher privacy protection degree reduces the prediction accuracy of the multi-label recognition model.

In order to alleviate the side-effect of additive Laplace noise, weight decay mechanism Zhang et al. (2018) inspires us to increase the prediction accuracy by reducing W^TW for the purse of improving P2-ML-GCN's robustness. Accordingly, we propose our model RP2-ML-GCN by integrating P2-ML-GCN's loss function with a regularization term as shown in Eq. (3.14).

$$\begin{aligned} L_{RP2} = &\sum_{i=1}^{C} y_i \log(\sigma(\hat{y}_i + Lap(0, \frac{S_{\hat{y}}}{\epsilon}))) \\ &+ (1 - y_i) \log(1 - \sigma(\hat{y}_i + Lap(0, \frac{S_{\hat{y}}}{\epsilon}))) + \lambda ||W||_2^F, \end{aligned} \tag{3.14}$$

where λ is a hyperparameter to control the weight of the regularization term, and the Forbenius norm $||W||_2^F$ is equal to the value of W^TW.

During the training process in P2-ML-GCN, we accomplish image recognition with privacy guarantee by minimizing L_{RP2} and improve the model's robustness by minimizing W^TW. Notably, in fact, the regularization term can improve the robustness of the traditional ML-GCN model even without additional noise.

3.4.2.3 Robustness Analysis

In the following, we theoretically investigate how the regularization term can improve P2-ML-GCN's robustness from two aspects. On the one hand, the regularization term helps shrink the space of weights so as to avoid overfitting. On the other hand, the utilization of the regularization term can reduce the variance of weights.

The training process of P2-ML-GCN can be treated as a generalized linear regression without regularization, while the training process of RP2-ML-GCN can be treated as a generalized ridge regression with regularization. Let W_{LR} and W_{Ridge} denote the weight matrixes trained in P2-ML-GCN and RP2-ML-GCN, respectively, which can be computed in Eq. (3.15) and Eq. (3.16), respectively.

$$W_{LR} = \underset{\boldsymbol{W}}{\text{argmin}} ||\hat{y}' - Wx||^2. \tag{3.15}$$

$$W_{Ridge} = \underset{\boldsymbol{W}}{\text{argmin}} ||\hat{y}' - Wx||^2 + \lambda ||W||^2. \tag{3.16}$$

Assume that x is centralized and standardized, and xx^T is reversible. We can obtain two estimators, *i.e.,* $\hat{W}_{LR}$ for W_{LR} and $\hat{W}_{Ridge}$ for W_{Ridge}.

$$\hat{W}_{LR} = \hat{y}' x^T (xx^T)^{-1}. \tag{3.17}$$

$$\begin{aligned}
\hat{W}_{Ridge} &= \hat{y}' x^T (xx^T + \lambda \mathbf{I})^{-1} \\
&= \hat{y}' x^T (xx^T)^{-1} (xx^T)(xx^T + \lambda \mathbf{I})^{-1} \\
&= \hat{W}_{LR} (xx^T)(xx^T + \lambda \mathbf{I})^{-1} \\
&= \hat{W}_{LR} (xx^T + \lambda \mathbf{I} - \lambda \mathbf{I})(xx^T + \lambda \mathbf{I})^{-1} \\
&= \hat{W}_{LR} (\mathbf{I} - \lambda (xx^T + \lambda \mathbf{I})^{-1}) \\
&\leq \hat{W}_{LR}.
\end{aligned} \tag{3.18}$$

Remark: From Eq. (3.17) and Eq. (3.18), $\hat{W}_{Ridge}$ can be considered as the shrinkage of $\hat{W}_{LR}$, achieving weight decay to avoid overfitting.

Lemma 3. *If $\hat{\mathcal{V}}$ is the unbiased estimator of any one random variable V, $\mathbb{E}(\hat{V}) = V$.*

Lemma 4. *Three numerical characteristics in matrix theory are shown as follows:*

$$\mathbb{E}(o^T Go) = (\mathbb{E}(o))^T G\mathbb{E}(o) + tr(G\,\mathrm{Var}(o)),$$

$$tr(EFG) = tr(FEG) = tr(GEF),$$

$$tr(G^T) = tr(G),$$

where o is white noise, and E, F and G represent any matrix.

Furthermore, to demonstrate that the regularization term indeed improves the model's robustness, we need to prove that the variance of $\hat{W}_{Ridge}$ is lower than the variance of $\hat{W}_{LR}$. Let $\hat{y}' = Wx + o$, where o is the white noise following $\mathcal{N}(0, \sigma^2)$. We rewrite $\hat{W}_{LR}$ in Eq.(3.19).

$$\begin{aligned} \hat{W}_{LR} &= \hat{y}' x^T (xx^T)^{-1} \\ &= (Wx + o)x^T (xx^T)^{-1} \\ &= W_{LR} + ox^T (xx^T)^{-1}. \end{aligned} \tag{3.19}$$

Since $\hat{W}_{LR}$ is an unbiased estimator, the variance of $\hat{W}_{LR}$ can be calculated using Lemma 3 and Lemma 4 as follows:

$$\begin{aligned}
\operatorname{Var}(\hat{W}_{LR}) &= \mathbb{E}(\hat{W}_{LR} - \mathbb{E}(\hat{W}_{LR}))^2 \\
&= \mathbb{E}(\hat{W}_{LR} - W_{LR})^2 \\
&= \mathbb{E}[(\hat{W}_{LR} - W_{LR})^T(\hat{W}_{LR} - W_{LR})] \\
&= \mathbb{E}[(ox^T(xx^T)^{-1})^T ox^T(xx^T)^{-1}] \\
&= \mathbb{E}[((xx^T)^{-1})^T xo^T ox^T(xx^T)^{-1}] \qquad (3.20) \\
&= \sigma^2 tr(((xx^T)^{-1})^T xx^T(xx^T)^{-1}) \\
&= \sigma^2 tr[((xx^T)^{-1})]^T \\
&= \sigma^2 tr((xx^T)^{-1}) \\
&= \sigma^2.
\end{aligned}$$

Similarly, the variance of $\hat{W}_{Ridge}$ can be calculated by:

$$\begin{aligned}
\operatorname{Var}(\hat{W}_{Ridge}) &= \sigma^2 \left[\sum_{i=1}^{\mathcal{K}} \frac{k_i}{(k_i+\lambda)^2}\right] \\
&= \left[\sum_{i=1}^{\mathcal{K}} \frac{k_i}{(k_i+\lambda)^2}\right] \operatorname{Var}(\hat{W}_{LR}) \qquad (3.21) \\
&= z \operatorname{Var}(\hat{W}_{LR}),
\end{aligned}$$

where $\mathcal{K}$ is the rank of xx^T, $(k_1, k_2, \cdots, k_{\mathcal{K}})$ is the set of eigenvalues of xx^T, and $z = \left[\sum_{i=1}^{\mathcal{K}} \frac{k_i}{(k_i+\lambda)^2}\right]$ denotes variance expansion factor.

Remark: The variance of $\hat{W}_{Ridge}$ can be lower than $\hat{W}_{LR}$ by adjusting λ. On the other hand, variance expansion factor z becomes smaller when the value of λ is increased, which further reduces the variance of $\hat{W}_{Ridge}$. Therefore, a conclusion can be drawn that RP2-

ML-GCN indeed improves the robustness of P2-ML-GCN by adding the regularization term from the viewpoint of the linear regression.

3.4.3 Bound of Global Sensitivity

To improve the prediction accuracy of P2-ML-GCN, there are two methods: one is to enhance the model's robustness, and the other is to decrease excessive noise added into the prediction outputs. A regularization term in RP2-ML-GCN can improve the model's robustness. In this subsection, we show that an appropriate bound of global sensitivity in differential privacy mechanisms can alleviate excessive noise's side effect. Before introducing our method, we present a critical observation as follows.

Observation 1. *Most existing analyses on differential privacy mechanisms assume that the maximum contribution (*i.e. *the global sensitivity of query function) is fixed in advance. However, we may end up adding excessive noise for privacy protection due to some outliers in database, resulting in the reduction of prediction accuracy of learning models. Therefore, a bound of global sensitivity of query function can be set to mitigate the side effect of excessive noise, which can improve the model performance Amin et al. (2019).*

According to Observation 1, the calculation of global sensitivity, $S_{\hat{y}}$, in P2-ML-GCN is affected by the imbalanced distribution of outputs, causing excessive noise. Inspired by the idea of Amin et al. (2019), we set a bound factor, denoted by $S_b \in (0, 1)$, to mitigate excessive noise's side effect for the improvement of P2-ML-GCN's accuracy.

In the following, we reimplement differential privacy mechanism with a bounded global sensitivity to see how it works to improve P2-ML-GCN's accuracy. First, we substitute $S_{\hat{y}}$ with $S_b S_{\hat{y}}$. According to Theorem 1, the disturbed output function satisfies $\frac{\epsilon}{S_b}$-differential privacy that is called relaxed-differential privacy in this paper because $S_b \in (0,1)$. Second, we rewrite the bias of loss function by replacing $Lap(0, \frac{S_{\hat{y}}}{\epsilon})$ with $Lap(0, \frac{S_b S_{\hat{y}}}{\epsilon})$ in Eq. (3.13), which is shown in Eq. (3.22).

$$\mathbb{E}(\Delta L) = |-\frac{1}{8} \times \frac{2S_b}{\epsilon^2}| = \frac{S_b S_{\hat{y}}}{4\epsilon^2}. \tag{3.22}$$

Eq. (3.22) implies that we can indeed decrease the bias of loss function in P2-ML-GCN by reducing the value of S_b and thus improve the prediction accuracy of P2-ML-GCN.

Remark: To guarantee relaxed-differential privacy and improve prediction accuracy simultaneously, we can select an appropriate bound for the global sensitivity in P2-ML-GCN and RP2-ML-GCN based on the specific distribution of outputs to alleviate excessive noise's side effect.

3.4.4 Model Effectiveness

As aforementioned in Section 3.2, prior differential privacy-based privacy-preserving deep learning approaches either protect the model's weights or input features. Different from the state-of-the-art, in our proposed models, privacy-preserving mechanisms are applied to protect the model's outputs, which can prevent black-box attack. In this subsection, we theoretically prove that our proposed models are also able to ensure ϵ-differential privacy for the model's weights and input features.

3.4.4.1 Effectiveness for Weights' Differential Privacy

Since the outputs of ML-GCN are calculated by both the weights and the input features, the noise added into outputs will reflect on the weights of classifiers and features through a backward propagation training process. In order to find out how the disturbed output vector $\hat{y}'$ influences the parameter matrix, the feature vector x is supposed to be fixed. We can rewrite the disturbed output vector with the disturbed parameter matrix, denoted by W_α, in Eq. (3.23).

$$\hat{y}' = \hat{y} + \alpha = W_\alpha x, \tag{3.23}$$

where $\hat{y}$ is the original output vector, and α is the additional Laplace noise used in differential privacy mechanism.

Let $\gamma_1 = \max\{|x_i^{-1}|\}$ with x_i^{-1} being the i-th element in vector x^{-1}, where each element in x^{-1} is the reciprocal of the corresponding element in x. Then, we can obtain the inequality in Eq. (3.24).

$$\begin{aligned} W_\alpha &= (\hat{y} + \alpha)x^{-1} = \hat{y}x^{-1} + \alpha x^{-1} \\ &= W + \alpha x^{-1} \leq W + \gamma_1 \alpha. \end{aligned} \tag{3.24}$$

Let $\overline{W}$ be the maximum value of elements in W and $\overline{W_\alpha}$ be the maximum value of elements in W_α, and $\max\{\alpha\}$ be the maximum value of elements in α. According to Eq. (3.24), we have $\overline{W_\alpha} = \overline{W} + \gamma_1 \max\{\alpha\}$.

Theorem 2. *If the disturbed output vector $\hat{y}'$ satisfies ϵ-differential privacy, the disturbed parameter matrix W_α satisfies $\frac{\epsilon(\overline{W} + \max\{|x_i^{-1}|\}\max\{\alpha\})}{\max\{|x_i^{-1}|\}^2}$-differential privacy.*

Proof. $\Pr[\cdot]$ is commonly designed as Laplace distribution. Since α follows $Lap(0, \frac{S_{\hat{y}}}{\epsilon})$, the additional Laplace noise can be designed as $\gamma_1\alpha$, which follows $Lap(0, \frac{\gamma_1^2 S_{\hat{y}}}{\epsilon})$, for the disturbed weight matrix W_α according to Eq. (3.24). Thus, we have

$$\begin{aligned} \ln\frac{\Pr[W]}{\Pr[W_\alpha]} &= \ln\frac{\frac{\epsilon}{2\gamma_1^2 S_{\hat{y}}} e^{-\frac{\epsilon}{\gamma_1^2 S_{\hat{y}}}|W|}}{\frac{\epsilon}{2\gamma_1^2 S_{\hat{y}}} e^{-\frac{\epsilon}{\gamma_1^2 S_{\hat{y}}}|W_\alpha|}} \\ &= \frac{\epsilon}{\gamma_1^2 S_{\hat{y}}}(|W_\alpha| - |W|) \leq \frac{\epsilon(\overline{W} + \gamma_1 \max\{\alpha\})}{\gamma_1^2} \\ &= \frac{\epsilon(\overline{W} + \max\{|x_i^{-1}|\}\max\{\alpha\})}{\max\{|x_i^{-1}|\}^2}. \end{aligned} \tag{3.25}$$

That is, we can prove that the disturbed weight matrix W_α satisfies $\frac{\epsilon(\overline{W}+\max\{|x_i^{-1}|\}\max\{\alpha\})}{\max\{|x_i^{-1}|\}^2}$-differential privacy. □

3.4.4.2 Effectiveness for Features' Differential Privacy

Similarly, in order to find out how the disturbed output vector $\hat{y}'$ influences the feature vector, we assume that the parameter matrix W is fixed. The disturbed output vector is rewritten with the disturbed feature vector, denoted by x_α, in Eq. (3.26).

$$\hat{y}' = \hat{y} + \alpha = W x_\alpha. \tag{3.26}$$

Let $\gamma_2 = \max\{|W_{ij}^{-1}|\}$ where W_{ij}^{-1} is the element in i-th row and j-th column in matrix W^{-1}. We can obtain the inequality in Eq. (3.27).

$$\begin{aligned} x_\alpha &= W^{-1}(\hat{y} + \alpha) = W^{-1}\hat{y} + W^{-1}\alpha \\ &= x + W^{-1}\alpha \leq x + \gamma_2\alpha. \end{aligned} \tag{3.27}$$

Let $\overline{x}$ be the maximum value of elements in x and $\overline{x}_\alpha$ be the maximum value of elements in x_α. From Eq. (3.27), there is $\overline{x}_\alpha = \overline{x} + \gamma_2 \max\{\alpha\}$.

Theorem 3. *If the disturbed output vector $\hat{y}'$ satisfies ϵ-differential privacy, the disturbed feature vector x_α satisfies $\frac{\epsilon(\overline{x}+\max\{|W_{ij}^{-1}|\}\max\{\alpha\}))}{\max\{|W_{ij}^{-1}|\}^2}$-differential privacy.*

Proof. $\Pr[\cdot]$ is commonly designed as Laplace distribution. Since α follows $Lap(0, \frac{S_{\hat{y}}}{\epsilon})$, the additional Laplace noise can be designed as $\gamma_2\alpha$, which follows $Lap(0, \frac{\gamma_2^2 S_{\hat{y}}}{\epsilon})$, for the disturbed feature vector x_α according to Eq. (3.27). Then we can prove that the disturbed feature vector x_α satisfies $\frac{\epsilon(\overline{x}+\max\{|W_{ij}^{-1}|\}\max\{\alpha\}))}{\max\{|W_{ij}^{-1}|\}^2}$-differential privacy as follows:

$$\begin{aligned} \ln\frac{\Pr[x]}{\Pr[x_\alpha]} &= \ln\frac{\frac{\epsilon}{2\gamma_2^2 S_{\hat{y}}} e^{-\frac{\epsilon}{\gamma_2^2 S_{\hat{y}}}|x|}}{\frac{\epsilon}{2\gamma_2^2 S_{\hat{y}}} e^{-\frac{\epsilon}{\gamma_2^2 S_{\hat{y}}}|x_\alpha|}} \\ &= \frac{\epsilon}{\gamma_2^2}(|x_\alpha| - |x|) \leq \frac{\epsilon(\overline{x} + \gamma_2 \max\{\alpha\})}{\gamma_2^2} \\ &= \frac{\epsilon(\overline{x} + \max\{|W_{ij}^{-1}|\}\max\{\alpha\}))}{\max\{|W_{ij}^{-1}|\}^2}. \end{aligned} \tag{3.28}$$

□

Remark: As analyzed in priors works Xu et al. (2019); Abadi et al. (2016), $\overline{W}$ and $\overline{x}$ are finite values. Although in our proposed models, we implement differential privacy mechanism on the model's outputs, Theorem 2 and Theorem 3 show the effectiveness of our models to achieve ϵ-differential privacy for model's weights and input features, which provides a new direction to perform differential privacy in deep learning algorithms.

3.4.5 Model Generalization

To further illustrate that our proposed models can achieve any degree of differential privacy for model's weights or input features, we extend our theoretical analysis to a more general scenario, in which two corollaries can be directly derived from Theorem 2 and Theorem 3.

Let $\mathcal{P}$ and $\gamma_{\mathcal{P}}$ be two finite values representing two scale parameters for the design of differential privacy on the model's weights. In the training process of the multi-label image recognition model, we set $\gamma_{\mathcal{P}} \geq \max\{|x_i^{-1}|\}$ by controlling the feature extractor first. Then, we set $\overline{W_\alpha} \leq \gamma_{\mathcal{P}}^2 \mathcal{P}$ when updating parameters. Accordingly, we can obtain Corollary 1.

Corollary 1. *If the disturbed output vector $\hat{y}'$ satisfies ϵ-differential privacy with $\gamma_{\mathcal{P}} \geq \max\{|x_i^{-1}|\}$ and $\overline{W_\alpha} \leq \gamma_{\mathcal{P}}^2 \mathcal{P}$, the disturbed weight matrix W_α satisfies $\epsilon\mathcal{P}$-differential privacy.*

Proof. If $\gamma_{\mathcal{P}} \geq \max\{|x_i^{-1}|\}$, $Lap(0, \frac{\gamma_{\mathcal{P}}^2}{\epsilon})$ can be considered as the additional Laplace noise to disturb weight matrix W according to Eq. (3.24). If $\overline{W_\alpha} \leq \gamma_{\mathcal{P}}^2 \mathcal{P}$, we can prove that the disturbed weight matrix W_α satisfies $\epsilon\mathcal{P}$-differential privacy as below:

$$\begin{aligned}\ln \frac{\Pr[W]}{\Pr[W_\alpha]} &= \ln \frac{\frac{\epsilon}{2\gamma_{\mathcal{P}}^2} e^{-\frac{\epsilon}{\gamma_{\mathcal{P}}^2}|W|}}{\frac{\epsilon}{2\gamma_{\mathcal{P}}^2} e^{-\frac{\epsilon}{\gamma_{\mathcal{P}}^2}|W_\alpha|}} \\ &= \frac{\epsilon}{\gamma_{\mathcal{P}}^2}(|W_\alpha| - |W|) \leq \frac{\epsilon \gamma_{\mathcal{P}}^2 \mathcal{P}}{\gamma_{\mathcal{P}}^2} \\ &= \epsilon\mathcal{P}.\end{aligned} \tag{3.29}$$

□

Similarly, we use two finite values, $\mathcal{Q}$ and $\gamma_{\mathcal{Q}}$, to denote two scale parameters for the design of differential privacy on model's input features. In the training process, we set $\gamma_{\mathcal{Q}} \geq \max\{|W_{ij}^{-1}|\}$ when updating parameters and set $\overline{x_\alpha} \leq \gamma_{\mathcal{Q}}^2 \mathcal{Q}$ when extracting features. Then, we can obtain Corollary 2.

Corollary 2. *If the disturbed output vector $\hat{y}'$ satisfies ϵ-differential privacy with $\gamma_{\mathcal{Q}} \geq \max\{|W_{ij}^{-1}|\}$ and $\overline{x_\alpha} \leq \gamma_{\mathcal{Q}}^2 \mathcal{Q}$, the disturbed feature vector x_α satisfies $\epsilon\mathcal{Q}$-differential privacy.*

Proof. If $\gamma_{\mathcal{Q}} \geq \max\{|W_{ij}^{-1}|\}$, $Lap(0, \frac{\gamma_{\mathcal{Q}}^2}{\epsilon})$ can be treated as the additional Laplace noise to disturb feature vector x according to Eq. (3.27). If $\overline{x_\alpha} \leq \gamma_{\mathcal{Q}}^2 \mathcal{Q}$, the disturbed feature vector x_α satisfies $\epsilon\mathcal{Q}$-differential privacy, which is proved below.

$$\begin{aligned} \ln \frac{\Pr[x]}{\Pr[x_\alpha]} &= \ln \frac{\frac{\epsilon}{2\gamma_{\mathcal{Q}}^2} e^{-\frac{\epsilon}{\gamma_{\mathcal{Q}}^2}|x|}}{\frac{\epsilon}{2\gamma_{\mathcal{Q}}^2} e^{-\frac{\epsilon}{\gamma_{\mathcal{Q}}^2}|x_\alpha|}} \\ &= \frac{\epsilon}{\gamma_{\mathcal{Q}}^2}(|x_\alpha| - |x|) \leq \frac{\epsilon\gamma_{\mathcal{Q}}^2\mathcal{Q}}{\gamma_{\mathcal{Q}}^2} \\ &= \epsilon\mathcal{Q}. \end{aligned} \tag{3.30}$$

□

Remark: Since $\mathcal{P}$ and $\mathcal{Q}$ can be any finite real number, we can successfully protect the model's weights and input features with any degree of differential privacy by implementing ϵ-differential privacy mechanisms on the model's outputs in our proposed models. Moreover, Corollary 1 and Corollary 2 provide a guidance for the design of privacy-preserving deep learning algorithms in a general scenario.

3.5 Experiment and Analysis

In this section, comprehensive experiments are conducted to validate that our two proposed models, P2-ML-GCN and RP2-ML-GCN, can effectively accomplish multi-label image recognition while guaranteeing ϵ-differential privacy; especially, compared with P2-ML-GCN, RP2-ML-GCN can increase prediction accuracy. Besides, experiments are set up to confirm that our proposed regularization term indeed improves the performance of ML-GCN model even without Laplace noise. Moreover, we investigate the advantage of setting a proper bound of global sensitivity to increase the accuracy of P2-ML-GCN by fine-tuning different values of S_b. Finally, the effectiveness of our proposed models is further evaluated through a comparison with the state-of-the-art.

3.5.1 Experiment Settings

The datasets, performance metrics, and mechanism implementation in our experiments are described below. Our implementation codes can be found in `https://github.com/ahahnut/-R-P2-ML-GCN`.

3.5.1.1 Datasets

We report the experimental results on two benchmark multi-label image recognition datasets, including Voc2007 Everingham et al. (2010) and MS-COCO Lin et al. (2014). Notice that there are 20 categories of images in Voc2007 (*i.e.* $C = 20$) and 80 categories of images in MS-COCO (*i.e.* $C = 80$). According to the definition of global sensitivity in Eq. (3.4), the global sensitivity $S_{\hat{y}}$ is set as 20 when we use Voc2007 in our experiments and is set as 80 when we use MS-COCO in our experiments.

In machine learning training, the number of proper epochs for different datasets are different. According to the state-of-the-art, we train Voc2007 dataset with 40 epochs and defaultly train MS-COCO dataset with 20 epochs. Similarly, we need to set different values of ϵ to make sure ϵ-differential privacy guarantee when training different datasets. The value of ϵ, which is so-called "privacy budget", indicates the degree of privacy protection. More specifically, a smaller ϵ implies a higher privacy protection degree.

3.5.1.2 Performance Metrics

Typically, the average per-class precision (CP), recall (CR), F1 (CF1), the average overall precision (OP), recall (OR), F1 (OF1), and mean average precision (mAP) are adopted to quantify prediction performance Wang et al. (2016a); Ge et al. (2018a); Zhu et al. (2017). For a fair comparison, the prediction performance of top-3 labels is also evaluated using the above performance metrics Ge et al. (2018a); Zhu et al. (2017) represented by ($*$_3), where $*$ could be OP, OR, OF1, CP, CR, CF1.

3.5.1.3 Mechanism Implementation

For clear performance evaluation, ML-GCN is adopted as the baseline model in our experiments. Our proposed models, including P2-ML-GCN and RP2-ML-GCN, are implemented according to the instructions of ML-GCN Chen et al. (2019b). There are four main steps in our experiments:

1. The dimensions of output features in two GCN layers are 1024 and 2048, respectively.

2. Label representations in GCN are adopted for training on Wikipedia dataset Pennington et al. (2014).

3. Resnet-101 He et al. (2016) is utilized to extract features of images resized into 448×448.

4. The parameter ϵ in Laplace noise is used to adjust the degree of privacy protection for privacy-preserving training.

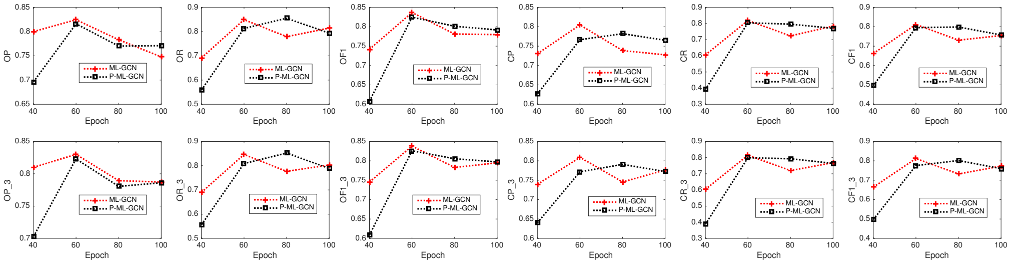

Figure 3.1 P2-ML-GCN v.s. ML-GCN on Voc2007 with $\epsilon = 8$

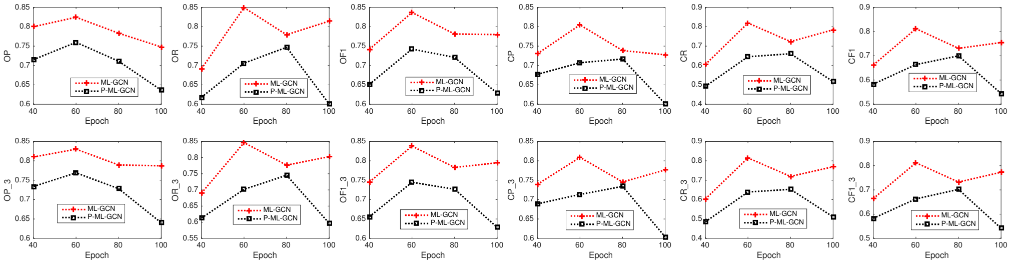

Figure 3.2 P2-ML-GCN v.s. ML-GCN on Voc2007 with $\epsilon = 10$

3.5.2 Evaluation of Privacy Preservation

We implement our model P2-ML-GCN with $\epsilon = 8, 10, 30$ in the experiments, which is reasonable and applicable in real applications according to the scenario of our studied problem and the setting of ϵ in previous works Abadi et al. (2016); Beaulieu-Jones et al. (2019); Xia

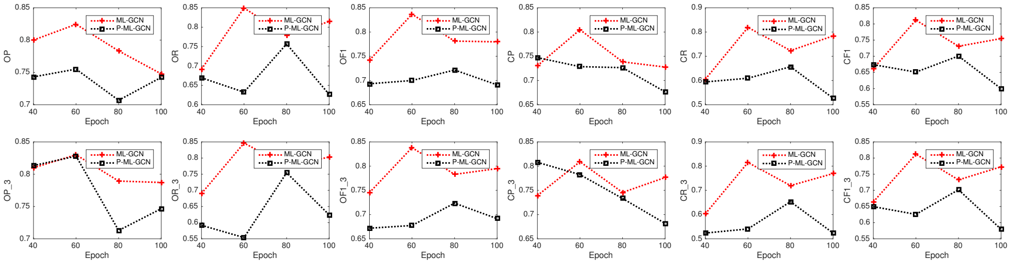

Figure 3.3 P2-ML-GCN v.s. ML-GCN on Voc2007 with $\epsilon = 30$

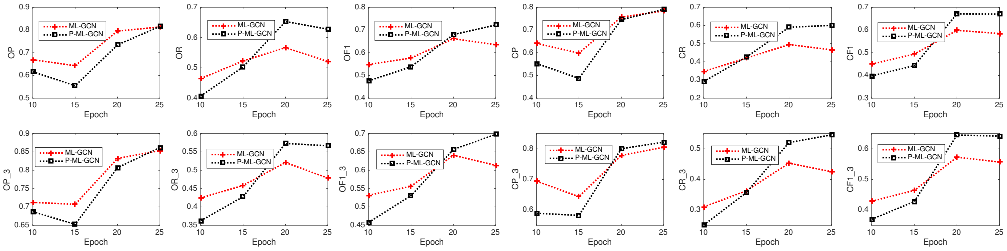

Figure 3.4 P2-ML-GCN v.s. ML-GCN on MS-COCO with $\epsilon = 8$

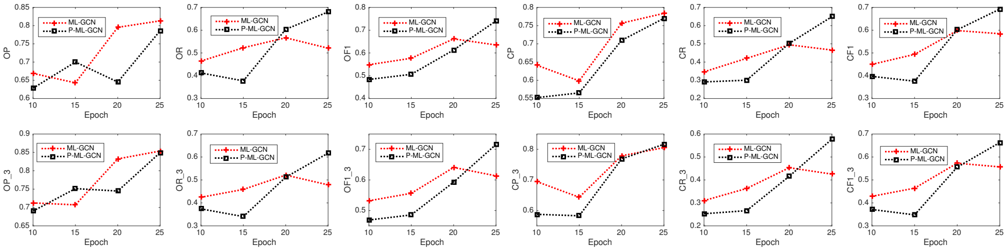

Figure 3.5 P2-ML-GCN v.s. ML-GCN on MS-COCO with $\epsilon = 10$

et al. (2019); Yoon et al. (2019). To illustrate the feasibility of P2-ML-GCN, the results on Voc2007 dataset with 40, 60, 80, and 100 epochs are presented in Fig. 3.1-Fig. 3.3, and the results on MS-COCO dataset with 10, 15, 20, and 25 epochs are presented in Fig. 3.4-Fig. 3.6. In these figures, obviously, P2-ML-GCN can achieve different degrees of ϵ-differential privacy guarantee by adjusting the values of ϵ; especially, a lower ϵ indicates a higher degree

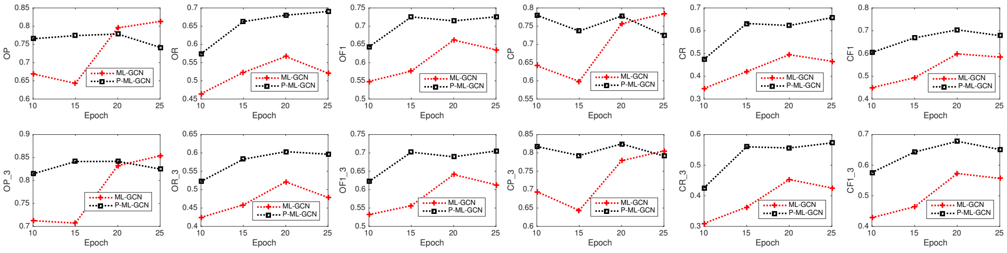

Figure 3.6 P2-ML-GCN v.s. ML-GCN on MS-COCO with $\epsilon = 30$

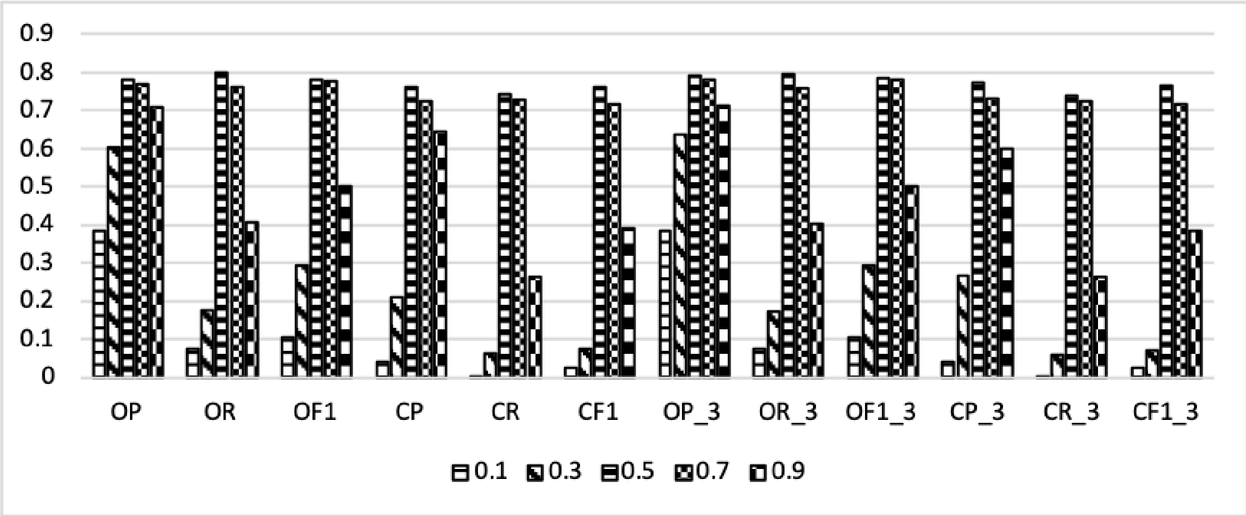

Figure 3.7 RP2-ML-GCN on Voc2007 with Different λ (Epoch:40; $\epsilon = 10$)

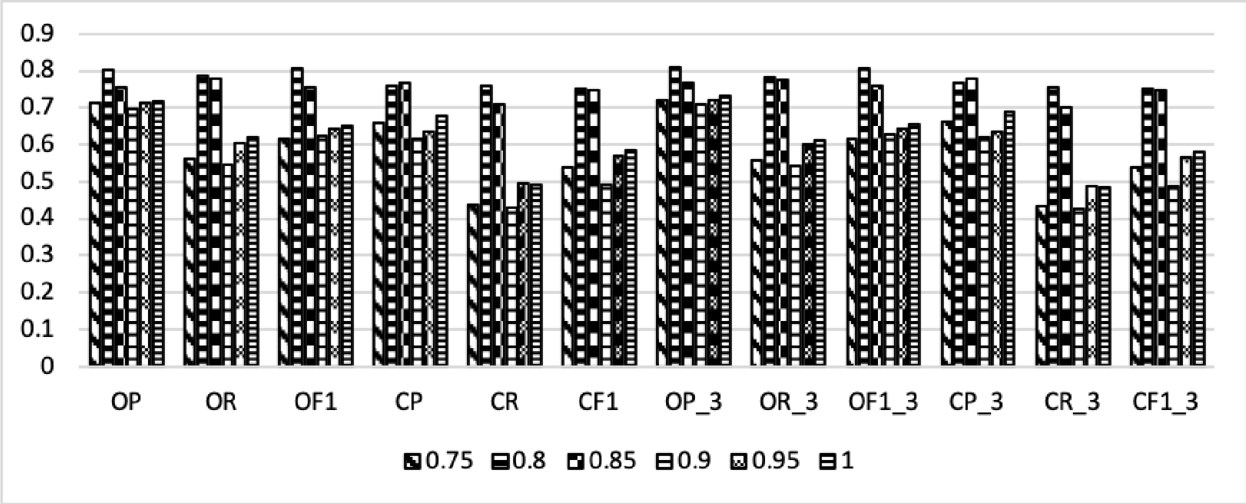

Figure 3.8 P2-ML-GCN with S_b on Voc2007 (Epoch:40; $\epsilon = 10$; Different S_b)

of privacy protection. In specific, take the OP value of P2-ML-GCN in Fig. 3.1 as an example. For P2-ML-GCN on Voc2007 dataset with 40 epochs, $OP = 0.6963$ in P2-ML-GCN with $\epsilon = 8$, $OP = 0.7152$ in P2-ML-GCN with $\epsilon = 10$, and $OP = 0.7431$ in P2-ML-GCN with $\epsilon = 30$. By comparing these OP values, we can find that the increase of the added

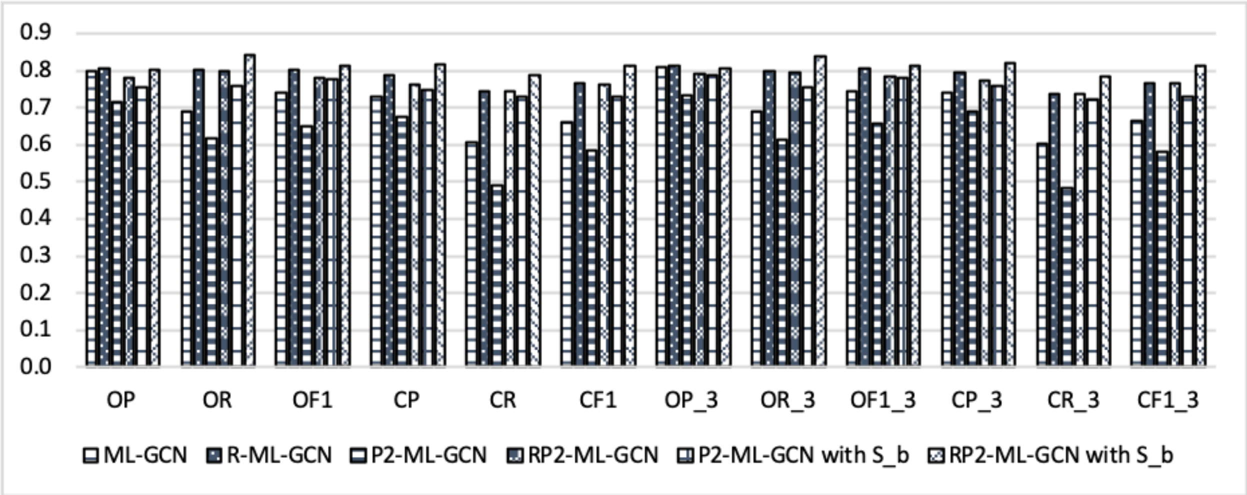

Figure 3.9 Evaluation Results on Voc2007 (Epoch: 40; $\epsilon = 10$; $\lambda = 0.5$; $S_b = 0.8$)

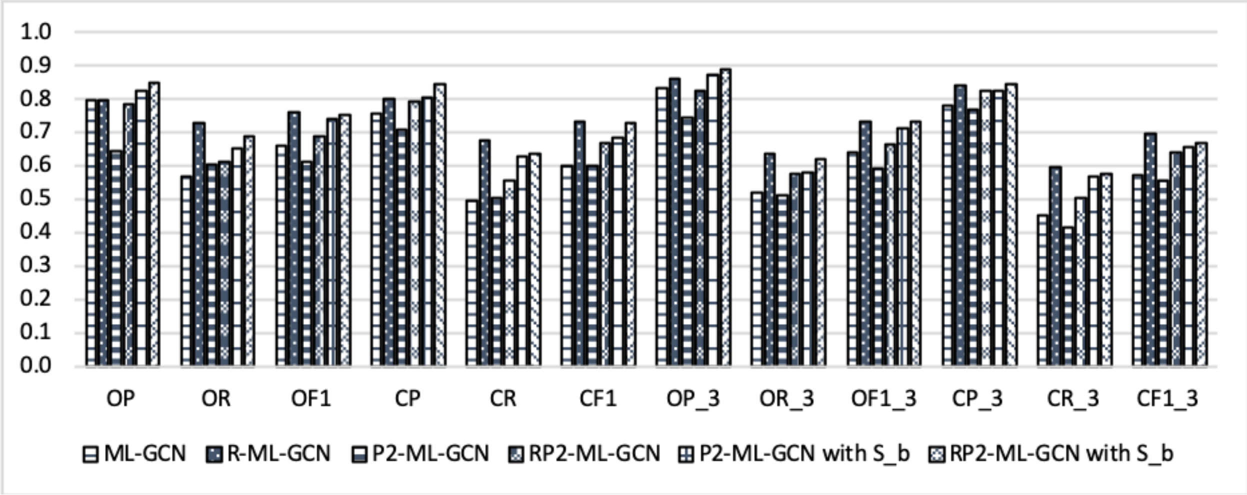

Figure 3.10 Evaluation Results on MS-COCO (Epoch: 20; $\epsilon = 10$; $\lambda = 0.5$; $S_b = 0.8$)

Laplace noise does not cause too much decrease of prediction performance of P2-ML-GCN. The same conclusion can be obtained by comparing other performance metrics on Voc2007 dataset in Fig. 3.1-Fig. 3.3. Besides, we can also get the same conclusion by comparing all performance metrics on MS-COCO dataset in Fig. 3.4-Fig. 3.6. To sum up, compared

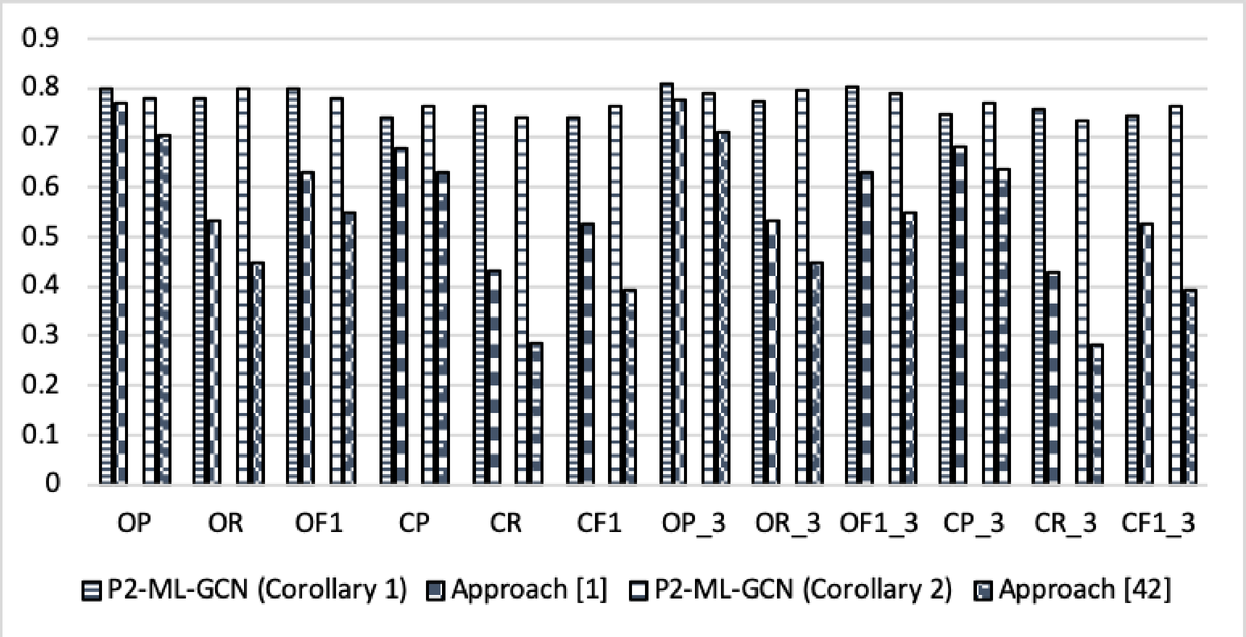

Figure 3.11 P2-ML-GCN on Voc2007 Satisfying 1-differential privacy (Epoch:40)

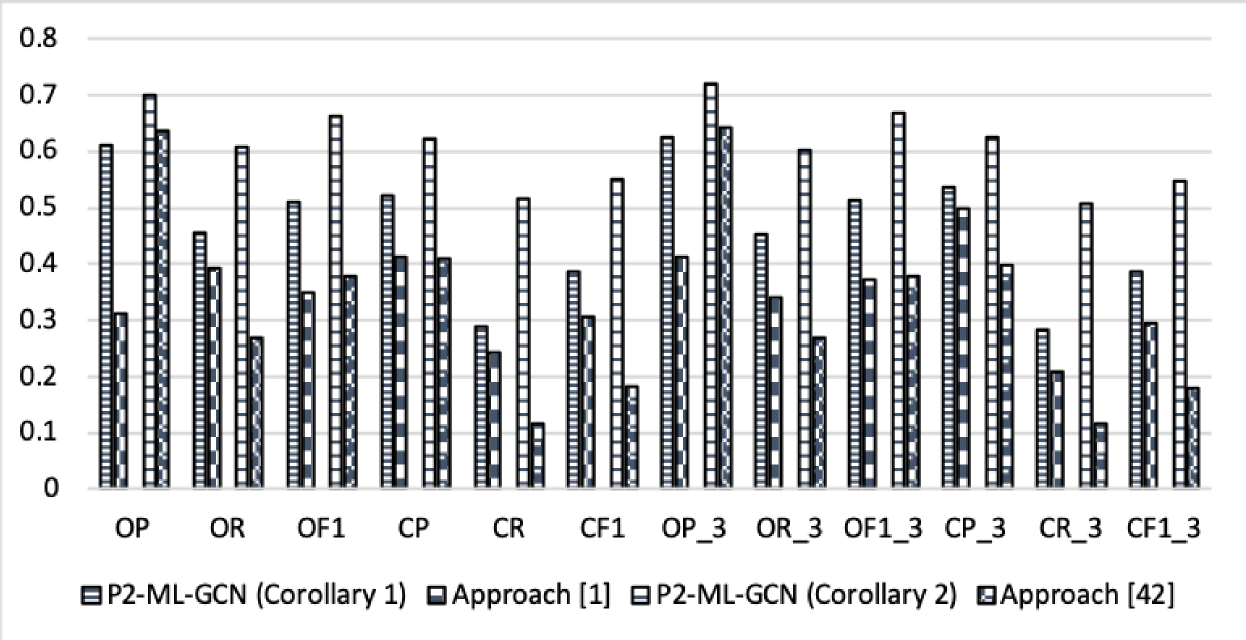

Figure 3.12 P2-ML-GCN on Voc2007 Satisfying 0.1-differential privacy (Epoch:40)

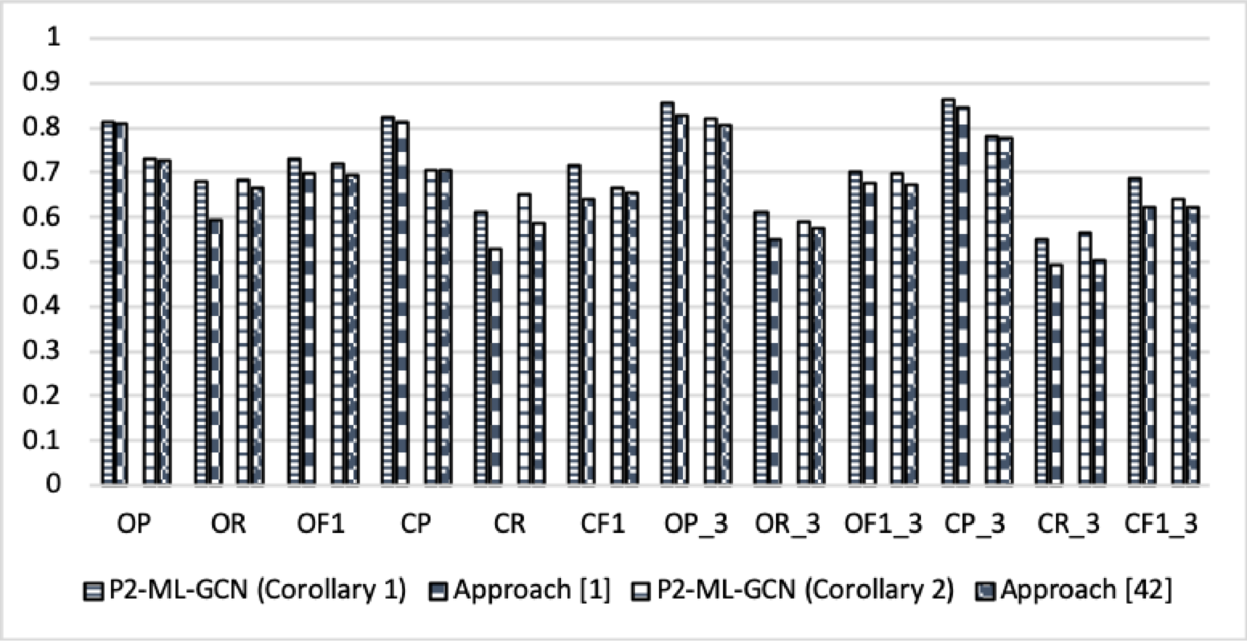

Figure 3.13 P2-ML-GCN on MS-COCO Satisfying 1-differential privacy (Epoch:20)

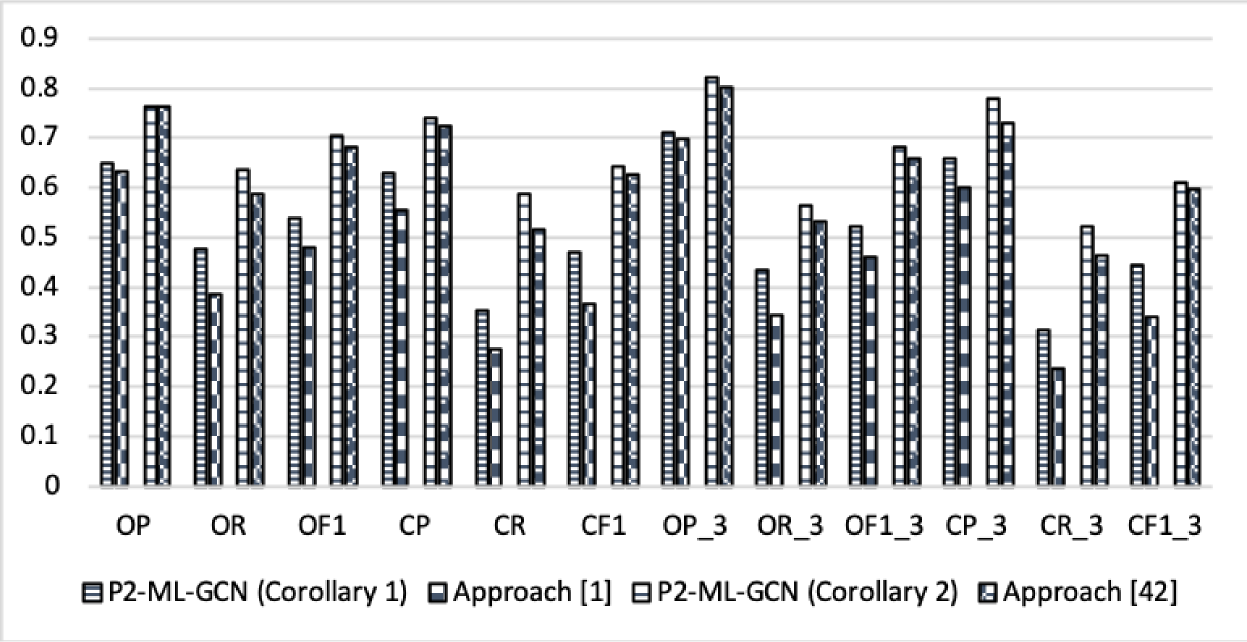

Figure 3.14 P2-ML-GCN on MS-COCO Satisfying 0.1-differential privacy (Epoch:20)

with ML-GCN, the prediction performance of P2-ML-GCN does not suffer a lot with the increase of the added Laplace noise, which indicates that P2-ML-GCN can maintain the performance of multi-label image recognition while providing ϵ-differential privacy guarantee. These results demonstrate the effectiveness of P2-ML-GCN for privacy protection as analyzed in subsection 3.4.1.

3.5.3 Ablation Study

We analyze that the scale parameter λ can be used to reduce the variance of loss function in Section 3.4.2.3, and the bounded global sensitivity $S_b S_{\hat{y}}$ can be used to decrease the bias of loss function in Section 3.4.3. Thus, in order to validate the analysis of the regularization term and the bounded global sensitivity, we present the following experiment results.

In Fig. 3.7, RP2-ML-GCN is trained on Voc2007 dataset with 40 epochs by fixing $\epsilon = 10$ and varying λ from 0.1 to 0.9 with 0.2 step size. We change the value of λ that represents the weight of the regularization term to observe its impact on the prediction performance in RP2-ML-GCN. Specifically, we use the OP value as an example to analyze this impact. In Fig. 3.7, $OP = 0.3835$ when $\lambda = 0.1$, $OP = 0.6022$ when $\lambda = 0.3$, $OP = 0.7793$ when $\lambda = 0.5$, $OP = 0.7702$ when $\lambda = 0.7$, and $OP = 0.7083$ when $\lambda = 0.9$. From these OP values, one can find that the OP value of RP2-ML-GCN can be highly improved when λ is increased from 0.1 to 0.5 but gradually decreases when λ is increased from 0.5 to 0.9. The same trend can also be observed by comparing other performance metrics. These phenomenons illustrate that the regularization term can be used to reduce the variance of loss function by adjusting the scale parameter λ as mentioned in Section 3.4.2.3. We will use $\lambda = 0.5$ when implementing

RP2-ML-GCN model in the following experiments.

In Fig. 3.8, P2-ML-GCN is trained on Voc2007 dataset with 40 epochs by fixing $\epsilon = 10$ and changing S_b from 0.75 to 1. The results show that with the same ϵ, different values of S_b can indeed affect the performance of P2-ML-GCN. From Fig. 3.8, we observe that $OP = 0.7134$ when $S_b = 0.75$, $OP = 0.8032$ when $S_b = 0.8$, $OP = 0.7555$ when $S_b = 0.85$, $OP = 0.6988$ when $S_b = 0.9$, $OP = 0.7131$ when $S_b = 0.95$, and $OP = 0.7152$ when $S_b = 1$. By comparing these OP values, a proper bound factor (*i.e.* $S_b = 0.8$ in our experiments) can be used to improve the OP value of the original P2-ML-GCN. For the other performance metrics in Fig. 3.8, the utilization of the proper bound factor $S_b = 0.8$ can also improve other performance metrics of the original P2-ML-GCN. In other words, a proper bound can indeed help enhance P2-ML-GCN's prediction performance while ensuring relaxed-differential privacy, which confirms the advantage of using a proper bounded global sensitivity to increase P2-ML-GCN's accuracy. More concretely, Theorem 1 tells that the disturbed output vector satisfies $\frac{\epsilon}{S_b}$-differential privacy when a bound factor, S_b, is set to the global sensitivity. Thus, according to the observation in Fig. 3.8, we set the proper bound factor as $S_b = 0.8$ to design Laplace noise with ϵ in the following experiments.

3.5.4 Evaluation of Our Proposed Approaches

The comparison results for ML-GCN, R-ML-GCN, P2-ML-GCN, RP2-ML-GCN, P2-ML-GCN with S_b, and RP2-ML-GCN with S_b are shown in this section. These six models are implemented on Voc2007 dataset with 40 epochs by setting $\epsilon = 10$, $\lambda = 0.5$ and $S_b = 0.8$, whose results are shown in Fig. 3.9. And they are also trained on MS-COCO dataset with

20 epochs by fixing $\epsilon = 10$, $\lambda = 0.5$ and $S_b = 0.8$, whose results are shown in Fig. 3.10.

The OP value is used as an example for analysis. In Fig. 3.9 and Fig. 3.10, $OP = 0.7152$ in P2-ML-GCN on Voc2007, $OP = 0.7793$ in RP2-ML-GCN on Voc2007, $OP = 0.6452$ in P2-ML-GCN on MS-COCO, and $OP = 0.7842$ in RP2-ML-GCN on MS-COCO. Obviously, the OP value of RP2-ML-GCN is higher than that of P2-ML-GCN on both two datasets. Also, from Fig. 3.9 and Fig. 3.10, we observe that RP2-ML-GCN's other performance metrics are higher than P2-ML-GCN's on both two datasets through simple comparison. All comparison results for P2-ML-GCN and RP2-ML-GCN demonstrate that RP2-ML-GCN can improve P2-ML-GCN's prediction performance by reducing the bias of loss function with the help of an additional regularization term, which is consistent with our theoretical analysis in subsubsection 3.4.2.2. In order to clearly illustrate the effectiveness of our proposed regularization term, we incorporated the regularization term into ML-GCN without adding Laplace noise, which is named R-ML-GCN. Concretely, we have $OP = 0.8001$ in ML-GCN on Voc2007, $OP = 0.8055$ in R-ML-GCN on Voc2007, $OP = 0.7954$ in ML-GCN on MS-COCO, and $OP = 0.7966$ in R-ML-GCN on MS-COCO, showning that R-ML-GCN's OP is higher than ML-GCN's on both datasets. Additionally, notice that R-ML-GCN's other performance metrics are higher than ML-GCN's on both two datasets. All these comparison results for ML-GCN and R-ML-GCN confirm that the regularization term can improve the prediction performance of ML-GCN even without Laplace noise, which has been analyzed in subsubsection 3.4.2.2.

Similarly, by observing Fig. 3.9 and Fig. 3.10, we obtain $OP = 0.7555$ in P2-ML-GCN with S_b on Voc2007, $OP = 0.7152$ in P2-ML-GCN on Voc2007, $OP = 0.8231$ in P2-ML-GCN with S_b on MS-COCO, and $OP = 0.6452$ in P2-ML-GCN on MS-COCO, which indicates that the OP value of P2-ML-GCN with S_b is better than that of P2-ML-GCN on both two datasets. In addition, P2-ML-GCN with S_b is better than P2-ML-GCN in terms of other performance metrics on both two datasets. Thus, we can improve the prediction performance of P2-ML-GCN by setting a proper bound to avoid excessive noise as analyzed in Section 3.4.3. Moreover, we train RP2-ML-GCN with S_b by integrating a regularization term and a proper bounded global sensitivity. Particularly, in Fig. 3.9, $OP = 0.8011$ in RP2-ML-GCN with S_b on Voc2007, $OP = 0.7152$ in P2-ML-GCN on Voc2007, $OP = 0.7793$ in RP2-ML-GCN on Voc2007, and $OP = 0.7555$ in P2-ML-GCN with S_b on Voc2007; in Fig. 3.10, we have $OP = 0.8491$ in RP2-ML-GCN with S_b on MS-COCO, $OP = 0.6452$ in P2-ML-GCN on MS-COCO, $OP = 0.7842$ in RP2-ML-GCN on MS-COCO, and $OP = 0.8231$ in P2-ML-GCN with S_b on MS-COCO. From these OP values, we can conclude that RP2-ML-GCN with S_b outperforms P2-ML-GCN, RP2-ML-GCN, and P2-ML-GCN with S_b on both two datasets. Besides, we can obtain the same conclusion by comparing other performance metrics. That is, RP2-ML-GCN with S_b is the best privacy-preserving deep learning model for multi-label image recognition among our proposed models.

3.5.5 Our Proposed Approaches v.s. *the State-of-the-Art*

According to Corollary 1, we set $\epsilon = 10$ in P2-ML-GCN and $\mathcal{P} = 1/10$, making the model's weights satisfy 1-differential privacy. Meanwhile, as indicated by Corollary 2, we set $\epsilon = 10$ in P2-ML-GCN and $\mathcal{Q} = 1/10$, making the model's input features achieve 1-differential privacy.

For a fair comparison, two existing schemes are adopted: (i) the scheme of Abadi et al. (2016) that adds the Laplace noise to make the model's weights meet ($\epsilon_w = 1$)-differential privacy; and (ii) the scheme of Yoon et al. (2019) that adds the Laplace noise to make the model's input features reach ($\epsilon_f = 1$)-differential privacy. The comparison results for 1-differential privacy on Voc2007 dataset and MS-COCO dataset are shown in Fig. 3.11 and Fig. 3.13, respectively. In a similar way, we conduct comparative experiments to make weights or input features satisfy 0.1-differential privacy, whose results are presented in Fig. 3.12 and Fig. 3.14.

For a clear illustration, we compare the OP values in Fig. 3.11. As shown in Fig. 3.11, we have $OP = 0.7995$ in P2-ML-GCN using Corollary 1 on Voc2007, $OP = 0.7714$ in the scheme of Abadi et al. (2016) on Voc2007, $OP = 0.7802$ in P2-ML-GCN using Corollary 2 on Voc2007, and $OP = 0.7063$ in the scheme of Yoon et al. (2019) on Voc2007. It can be seen that with the same degree of ϵ-differential privacy, the OP value of P2-ML-GCN is better than that of the two existing schemes. And via the same simple comparison, in Fig. 3.11, other performance metrics of P2-ML-GCN are also better than those of the two existing schemes. That is, with the same degree of ϵ-differential privacy, the prediction performance of P2-ML-GCN is better than that of the two existing schemes, indicating that our P2-ML-GCN model outperforms the two existing schemes. Additionally, we can also obtain the same conclusion by comparing the results of Fig. 3.12, Fig. 3.13, and Fig. 3.14. Furthermore, compared with P2-ML-GCN, RP2-ML-GCN can achieve the same degree of ϵ-differential privacy and enhanced prediction performance. Thus, we can conclude that

RP2-ML-GCN also outperforms the two existing schemes, for which the main reason is that the noise added to the model's outputs in both P2-ML-GCN and RP2-ML-GCN can be bounded in deep learning training even if a neural network contains many layers.

Evaluation Summary: All of the above experiments clearly demonstrate the superiority of our two proposed models, P2-ML-GCN and RP2-ML-GCN, in ensuring privacy protection, mitigating noise's side effect, and maintaining the model's accuracy, which is consistent with our theoretical analysis in Section 3.4.

3.6 Conclusion

In this paper, we firstly propose P2-ML-GCN model to achieve privacy guarantee while accomplishing multi-label image recognition. Then, the Forbenius norm of weights in GCN is designed as a regularization term in RP2-ML-GCN to improve the prediction accuracy and robustness of P2-ML-GCN. Additionally, the idea of bounded global sensitivity is exploited to enhance the prediction accuracy. In both P2-ML-GCN and RP2-ML-GCN, our privacy-preserving mechanism implemented on the model's outputs not only can defend black-box attack but also can provide privacy protection for the model's weights and input features. Moreover, the effectiveness of privacy protection, regularization term, and bounded global sensitivity in our proposed models has been rigorously proved. The results of comprehensive real-data experiments, especially the comparison with the state-of-the-art, can validate the advantages of our proposed models.

CHAPTER 4
PRIVACY-PRESERVING MULTIMODAL SENTIMENT ANALYSIS

4.1 Motivation

With the proliferation of social media, the importance of multimodal sentiment analysis has attracted the attention of researchers Mihalcea (2012); Poria et al. (2020) for stock market performance prediction Bollen et al. (2011), election outcome prediction Tumasjan et al. (2010), customer satisfaction assessment and brand perception analysis Jansen et al. (2009), and human-computer interaction Rahmani et al. (2021); Cai et al. (2021a). Nowadays, driven by the explosive progress of deep learning technology, learning-based prediction has been treated as one promising and effective approach to realize multimodal sentiment analysis through multimodal data representations extracted from raw multimedia data Bengio et al. (2013); Khorram et al. (2018); Piersol & Beddingfield (2019); Pang et al. (2020). Unfortunately, such extracted data representations can be exploited to infer private information (*e.g.*, user identification and location) by malicious attackers, causing serious privacy threats and substantial economic loss to individuals Hajian & Domingo-Ferrer (2012). Therefore, *how to protect individual data privacy in multimodal sentiment analysis becomes an important issue to be solved urgently.*

In order to prevent privacy leakage from learning-based multimodal sentiment analysis methods, a number of privacy-preserving learning algorithms have been proposed. One vein of research is based on adversarial training to generate adversarial samples that is used as the data disturbed by noise to defend inference attacks not only on unimodal data Li et al. (2020c); Ding et al. (2020) but also on multimodal data Jaiswal & Provost (2020); Xiong et al. (2021b); Xu et al. (2021); Huang et al. (2023); Xiong et al. (2023a). Although these adversarial training-based models are widely applied to privacy-preserving learning schemes, they fail to provide any performance guarantee of data privacy protection. Differential privacy-based models have been developed to guarantee data privacy protection by disturbing the data via the addition of Laplace noise based on differential privacy mechanisms Chamikara et al. (2020); Wang et al. (2016b, 2013); Xu et al. (2022b). However, it is worth mentioning that the data correlation can be treated as side-channel information, thus reducing the effectiveness of differential privacy protection. As a result, for correlated data, the additional Laplace noise used in differential privacy mechanisms should be enlarged with the increase of data correlation to maintain the same differential privacy protection degree, inevitably sacrificing the learning performance (*e.g.*, accuracy) Hu & Yang (2020); Ou et al. (2016); Zhang et al. (2019b). Furthermore, to mitigate the impact of data correlation on performance loss, the existing differentially private transform-based approaches transform the correlated homogeneous data into the corresponding uncorrelated data domain and then implement differential privacy mechanisms to achieve data privacy guarantee Wang et al. (2021); Rastogi & Nath (2010); Xiao et al. (2010); Jiang et al. (2016). Nevertheless, these existing transform-based approaches can only perform the transformation on homogeneous data with intra-correlation (that means data correlation within a data instance, such as temporal correlation in a video and location correlation in a trajectory) but are not applicable to heterogeneous data with inter-correlation (that means correlation among different data instances, such as data correlation between two texts and data correlation between a video and an audio). This is because the transformation schemes in the previous works, including Discrete Fourier Transform (DFT), Wavelet Transform (WT), and Principle Component Analysis (PCA), can only process the correlated homogeneous data to generate uncorrelated representations. Therefore, it is still a challenging task to generate privacy-preserving representations of the correlated heterogeneous multimodal data while maintaining the performance of multimodal sentiment analysis.

Motivated by the above analysis, in this paper, we devise a novel model, named **Differentially Private Correlated Representation Learning (DPCRL)**, to generate privacy-preserving multimodal representations for multimodal sentiment analysis by integrating a correlated representation learning scheme and a differential privacy protection scheme. The correlated representation learning scheme is designed as a heterogeneous multimodal data transformation strategy to learn the correlated and uncorrelated multimodal representations, in which a correlated factor can be pre-determined to flexibly adjust the expected correlation among the correlated multimodal representations. The differential privacy protection scheme is further applied to generating the disturbed correlated and uncorrelated representations by adding Laplace noise for satisfying ϵ-differential privacy. More specifically, a proper correlation factor can be set in our DPCRL model to extract the correlated representations with a relatively lower correlation, thus mitigating the side-effect of the additional Laplace noise on sentiment prediction performance. Finally, we evaluate the effectiveness of our DPCRL model on real-world datasets by conducting comprehensive experiments. Our multifold contributions are addressed as follows.

- To the best of our knowledge, this is the first work to design privacy-preserving multimodal sentiment analysis model.
- Our proposed DPCRL model seamlessly combines a correlated representation learning scheme with a differential privacy protection scheme, aiming at simultaneously ensuring ϵ-differential privacy and retaining the performance of multimodal sentiment analysis.
- In our correlated representation learning scheme, the heterogeneous multimodal data transformation can be accomplished by learning the correlated and uncorrelated multimodal representations from multimodal data for sentiment prediction, and the expected correlation of correlated representations can be flexibly set via a correlation factor.
- Comprehensive experiments are well conducted to validate the advantages of our DPCRL model over the state of the art for privacy-preserving multimodal sentiment analysis.

The rest of this paper is organized as follows. The related works are briefly summarized in Section 4.2. We elaborate the details of our model in Section 4.3, and then conduct real-data experiments and analyze all the results in Section 4.4. Finally, we end up with a conclusion in Section 4.5.

4.2 Related Works

In this section, we summarize the related works on multimodal sentiment analysis and review the current mainstream privacy-preserving learning approaches.

4.2.1 Multimodal Sentiment Analysis

The methodology of multimodal sentiment analysis can be broadly divided into two categories, including utterance-level models and inter-utterance contextual models. (i) **Utterance-level models** focus on an utterance to analyze multimodal sentiment. In general, utterance-level models devote to designing sophisticated fusion mechanisms, including decision-level fusion Poria et al. (2015, 2016) and feature-level fusion Liu et al. (2018); Mai et al. (2019a,b); Zadeh et al. (2017); Hazarika et al. (2020), for multimodal sentiment analysis. Moreover, multimodal-aware word embeddings Chen et al. (2017); Wang et al. (2019d), graph-based fusion Mai et al. (2020); Zadeh et al. (2018c), memory and attention mechanisms Zadeh et al. (2018a,b) have been considered to outperform representations fusion from a more fine-grained view. (ii) **Inter-utterance contextual models** take neighboring utterances in an overall video into account. The inter-utterance contextual model was first proposed by Poria *et al.* to learn inter-utterance representations by formulating the sentiment analysis of videos as a sequence tagging task Poria et al. (2017a). Later, the inter-utterance contextual models have been utilized to improve fusion effect with the help of attention mechanisms Chauhan et al. (2019); Poria et al. (2017b); Chen & Luo (2019) and hierarchical fusion Majumder et al. (2018) as well as to develop better contextual models Akhtar et al. (2019); Ghosal et al. (2018); Gu et al. (2018b). In a nutshell, representation learning is the primary technical component in the multimodal sentiment analysis models. It is also the mainstream way to learn specific representations contained in each modality data and invariant representations shared in multimodal data for multimodal sentiment prediction Hazarika et al. (2020). However, no work has been proposed to capture the correlated and uncorrelated representations in multimodal data from the viewpoint of correlation.

4.2.2 Privacy-Preserving Learning Approaches

Currently, adversarial training-based models, differential privacy-based approaches, and differentially private transform-based methods are the mainly popular techniques used in machine learning for data privacy protection. (i) **Adversarial training-based models** are exploited to generate adversarial samples that are taken as the data disturbed by noise to defend learning-based inference attacks not only for unimodal data Li et al. (2020c); Ding et al. (2020); Xu et al. (2021); Li et al. (2020b) but also for multimodal data Jaiswal & Provost (2020); Xiong et al. (2021b); Xu et al. (2021). Although the adversarial training is relatively attractive to be employed in privacy-preserving learning schemes owing to its convenience and efficiency, it cannot ensure a privacy protection guarantee. (ii) **Differential privacy-based approaches** are proposed to provide a theoretical guarantee of data privacy protection by adding Laplace noise based on differential privacy mechanisms Wang et al. (2013, 2016b); Chamikara et al. (2020). In particular, for the correlated data, the added Laplace noise should be increased with the growth of data correlation so as to ensure the theoretical guarantee of data privacy protection Hu & Yang (2020); Ou et al. (2016); Zhang et al. (2019b), which however, sacrifices the performance (*e.g.*, accuracy) of learning models. (iii) **Differentially private transform-based methods** transform the correlated data into the corresponding uncorrelated data domain and then apply differential privacy mechanisms to preserve data privacy Wang et al. (2021); Rastogi & Nath (2010); Xiao et al. (2010); Jiang et al. (2016), where the side-effect of the larger Laplace noise on learning performance can be eliminated due to the disappearance of data correlation after data transformation. Unfortunately, these existing transform-based methods can only be used to transform the homogeneous data with intra-correlation into independent (uncorrelated) data domain but cannot be applied to the heterogeneous multimodal data with inter-correlation.

In this paper, a novel DPCRL model is proposed to ensure differential privacy while maintaining the performance of multimodal sentiment analysis. In DPCRL, the heterogeneous multimodal data transformation can be achieved by learning the correlated and uncorrelated multimodal representations, where especially, a pre-determined correlation factor can be used to adjust the expected correlation of the correlated representations. More importantly, a proper correlation factor can help mitigate the side-effect of the added Laplace noise on sentiment prediction performance.

4.3 Methodology

In this section, we elaborate on the details of our proposed DPCRL model. As shown in Fig. 4.1, the DPCRL model is made up of five components, including a feature extraction module, an encoding module, a decoding module, a differential privacy protection mod-

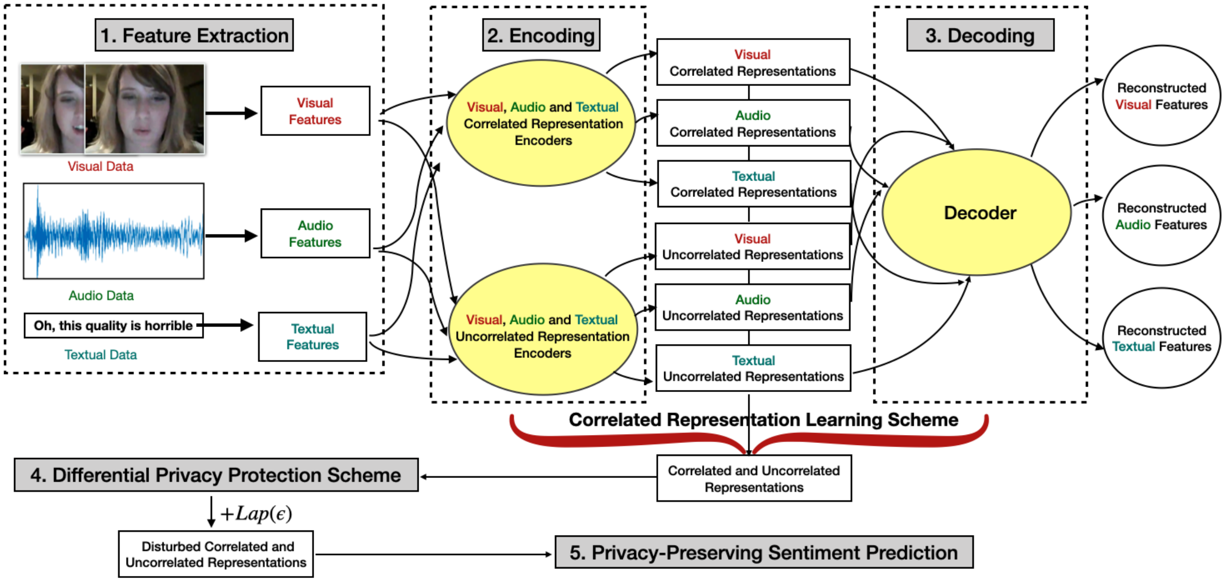
1. Feature Extraction
Visual Data
Audio Data
Oh, this quality is horrible
Textual Data
Visual Features
Audio Features
Textual Features
2. Encoding
Visual, Audio and Textual Correlated Representation Encoders
Visual, Audio and Textual Uncorrelated Representation Encoders
Visual Correlated Representations
Audio Correlated Representations
Textual Correlated Representations
Visual Uncorrelated Representations
Audio Uncorrelated Representations
Textual Uncorrelated Representations
3. Decoding
Decoder
Reconstructed Visual Features
Reconstructed Audio Features
Reconstructed Textual Features
Correlated Representation Learning Scheme
Correlated and Uncorrelated Representations
4. Differential Privacy Protection Scheme
$+Lap(\epsilon)$
Disturbed Correlated and Uncorrelated Representations
5. Privacy-Preserving Sentiment Prediction

Figure 4.1 The Data Flow of Our DPCRL Model

ule, and a privacy-preserving sentiment prediction module. Firstly, a feature extraction scheme is designed to extract features from video, audio and language modalities. Secondly, in the encoding module, we use the correlated and uncorrelated multimodal representation encoders to learn the correlated and uncorrelated multimodal representations from the extracted features, where a correlation factor is used in the correlated multimodal representation encoders to obtain the correlated multimodal representations. Thirdly, the decoding module is devised to reconstruct the extracted features by decoding the correlated and uncorrelated representations in each modality, which helps the encoding module avoid encoding the unrepresentative vector in each modality. This autoencoding architecture of the correlated representation learning actually works as a heterogeneous multimodal data transform scheme in DPCRL. Fourthly, a differential privacy protection scheme is leveraged to obtain privacy-preserving representations by adding Laplace noise to the correlated and uncorrelated representations learned from the previous autoencoding architecture. Finally, these perturbed representations are put into the privacy-preserving sentiment prediction module to accomplish the privacy-preserving multimodal sentiment analysis task.

For real-world implementation, the first four components in DPCRL should be deployed on the users' device side, and the last one should be implemented on the server side. When running DPCRL, the first four components are executed on the users' device side to generate the privacy-preserving representations, which will be transmitted to the server side for the final prediction using the last component. DPCRL can help users avoid privacy leakage caused by attackers who can leverage the eavesdropped representations during transmission to infer the raw users' sensitive data via some effective deep learning attack models, such as the membership inference attack and the inversion attack. In the following, we introduce these five modules in DPCRL one by one.

4.3.1 Feature Extraction

Each video is segmented into utterances, each of which is a unit of speech bounded by breaths or pauses Olson (1977). An utterance comprises a sequence of visual modality data denoted as $\mathbf{U}_v \in \mathbb{R}^{T_v \times d_v}$, a sequence of acoustic modality data denoted as $\mathbf{U}_a \in \mathbb{R}^{T_a \times d_a}$, and a sequence of language modality data denoted $\mathbf{U}_l \in \mathbb{R}^{T_l \times d_l}$, where T_m ($m \in \{v, a, l\}$) represents the length of an utterance, and d_m represents the number of dimensions of the modality data. For feature extraction, the stacked bi-directional Long Short-Term Memory scheme (sLSTM) Hyvärinen & Oja (1997) is exploited to map $\mathbf{U}_m \in \mathbb{R}^{T_m \times d_m}$ into a feature vector $\mathbf{f}_m \in \mathbb{R}^{d_h}$ $(m \in \{v, a, l\})$ with d_h being the size of hidden states set in the sLTSM model:

$$\mathbf{f}_m = sLSTM(\mathbf{U}_m; \theta_m^{slstm}), \tag{4.1}$$

where θ_m^{lstm} represents the parameters of sLSTM.

4.3.2 Encoding

In the encoding process, the visual/acoustic/language modality data is processed by taking into account the following three requirements: (i) for each feature vector $\mathbf{f}_m$ $(m \in \{v, a, l\})$, its correlated and uncorrelated representations should capture two distinctive aspects of the same modality data; (ii) any two of the uncorrelated representations of $\mathbf{f}_v$, $\mathbf{f}_a$, and $\mathbf{f}_l$ should be distinctive without redundancy; and (iii) the correlation between any two of the correlated representations of $\mathbf{f}_v$, $\mathbf{f}_a$, and $\mathbf{f}_l$ should be close to the correlation factor c as much as possible.

First of all, as shown by domain separation networks Bousmalis et al. (2016a), each feature vector $\mathbf{f}_m$ can be projected to two distinct types of representations. Thus, given $\mathbf{f}_m$, we use the correlated multimodal representation encoder E_m^c to extract the corresponding correlated representation $\mathbf{f}_m^c \in \mathbb{R}^{d_h}$ and employ the uncorrelated multimodal representation encoder E_m^u to capture the corresponding uncorrelated representation $\mathbf{f}_m^u \in \mathbb{R}^{d_h}$:

$$\mathbf{f}_m^c = E_m^c(\mathbf{f}_m; \theta_m^c, c), \tag{4.2}$$

$$\mathbf{f}_m^u = E_m^u(\mathbf{f}_m; \theta_m^u), \tag{4.3}$$

where θ_m^c represents the parameters of the encoder E_m^c, θ_m^u represents the parameters of the encoder E_m^u, and c represents an expected correlation factor that is set to obtain the correlated representations with the expected correlation.

According to Bousmalis et al. (2016b), the orthogonality constraint can be used to achieve non-redundancy between two representations. Therefore, to satisfy the first and the second requirements of encoding, we formulate the *data orthogonality loss*, $\mathcal{L}_{enc_1}$:

$$\mathcal{L}_{enc_1} = \sum_{m \in \{v,a,l\}} ||{\mathbf{f}_m^c}^T \mathbf{f}_m^u||_F^2 + \sum_{m \neq m' \in \{v,a,l\}} ||{\mathbf{f}_m^u}^T \mathbf{f}_{m'}^u||_F^2, \tag{4.4}$$

where $|| \cdot ||_F^2$ is the squared Frobenius norm.

Then, inspired by the idea of Sun et al. (2016), we use the cosine distance to quantify the correlation between two correlated representations. Considering the third requirement of encoding, we define the *data correlation loss*, $\mathcal{L}_{enc_2}$:

$$\mathcal{L}_{enc_2} = \sum_{m \neq m' \in \{v,a,l\}} ||{\mathbf{f}_m^c}^T \mathbf{f}_{m'}^c - cI||_F^2, \tag{4.5}$$

where $c \in [0, 1]$ is a correlation factor that indicates the cosine distance between two representations, and I denotes the identity matrix. To sum up, the entire encoding loss function $\mathcal{L}_{enc}$ is the summation of $\mathcal{L}_{enc_1}$ in Eq. (4.4) and $\mathcal{L}_{enc_2}$ in Eq. (4.5), shown in Eq. (4.6).

$$\mathcal{L}_{enc} = \mathcal{L}_{enc_1} + \mathcal{L}_{enc_2}. \tag{4.6}$$

4.3.3 Decoding

Since an encoder function may output an unrepresentative vector that cannot be recovered, we design a decoder D to reconstruct the original feature vector by using the extracted correlated and uncorrelated representations (*i.e.* $\mathbf{f}_m^c$ and $\mathbf{f}_m^u$) in each modality. The decoder D is defined in Eq. (4.7) to ensure that the encoded representations indeed represent the details of the corresponding modality data Hazarika et al. (2020); Bengio et al. (2013).

$$\overline{\mathbf{f}}_m = D(\mathbf{f}_m^c + \mathbf{f}_m^u; \theta_d), \tag{4.7}$$

where $\overline{\mathbf{f}}_m$ is the reconstructed feature vector for $m \in \{v, a, l\}$, and θ_d represents the parameters of the decoder D. In the decoding process, the reconstruction loss, $\mathcal{L}_{dec}$, is measured by *mean squared error* as below:

$$\mathcal{L}_{dec} = \sum_{m \in \{v,a,l\}} \frac{||\mathbf{f}_m - \overline{\mathbf{f}}_m||_2^2}{d_h}, \tag{4.8}$$

where $|| \cdot ||_2^2$ denotes the squared $L2$-norm.

Finally, the correlated representation learning can be achieved through the autoencoding architecture that is the combination of the encoders and the decoders. Correspondingly, the loss function of the correlated representation learning process, $\mathcal{L}_{CRL}$, is the summation of the encoding loss $\mathcal{L}_{enc}$ in Eq. (4.6) and the decoding loss $\mathcal{L}_{dec}$ in Eq. (4.8), *i.e.*,

$$\mathcal{L}_{CRL} = \alpha\mathcal{L}_{enc} + \beta\mathcal{L}_{dec}, \tag{4.9}$$

where $\alpha \in (0, 1]$ and $\beta \in (0, 1]$ are the weights of loss functions. We minimize $\mathcal{L}_{CRL}$ to obtain the correlated and uncorrelated multimodal representations for multimodal sentiment analysis.

4.3.4 Differential Privacy Protection Scheme

After obtaining the correlated and uncorrelated representations through our proposed correlated representation learning, we implement the differential privacy mechanisms to generate privacy-preserving representations for multimodal sentiment analysis. To be specific, in our differential privacy protection scheme, the representations captured by our proposed correlated representation learning and the privacy-preserving representations are considered as the neighboring databases in differential privacy theory. In the following, we apply different differential privacy mechanisms to the correlated and uncorrelated representations.

Firstly, according to *Basic Differential Privacy Mechanism* Dwork et al. (2006), we can calculate the perturbed uncorrelated representation $\hat{\mathbf{f}}_m^u = \mathbf{f}_m^u + Lap\left(0, S_{\mathbf{f}_m^u}/\epsilon\right)$ by using an additional Laplace noise to satisfy ϵ-differential privacy, where $S_{\mathbf{f}_m^u}$ represents the global sensitivity of the uncorrelated representation vector $\mathbf{f}_m^u$ and is equal to the difference between the maximal and the minimal items in $\mathbf{f}_m^u$.

Theorem 4. *Given the Laplace noise $Lap(0, S_{\mathbf{f}_m^u}/\epsilon)$ added into the uncorrelated representation vector $\mathbf{f}_m^u$, the disturbed uncorrelated representation vector $\hat{\mathbf{f}}_m^u$ satisfies ϵ-differential privacy.*

Proof. Let $\Pr[\cdot]$ be a commonly designed Laplace distribution Eltoft et al. (2006). Accordingly, we have

$$\ln \frac{\Pr[\mathbf{f}_m^u]}{\Pr[\hat{\mathbf{f}}_m^u]} = \ln \frac{\frac{\epsilon}{2S_{\mathbf{f}_m^u}} e^{-\frac{\epsilon}{S_{\mathbf{f}_m^u}}|\mathbf{f}_m^u|}}{\frac{\epsilon}{2S_{\mathbf{f}_m^u}} e^{-\frac{\epsilon}{S_{\mathbf{f}_m^u}}|\hat{\mathbf{f}}_m^u|}} = \frac{\epsilon}{S_{\mathbf{f}_m^u}}(|\hat{\mathbf{f}}_m^u| - |\mathbf{f}_m^u|) \leq \epsilon. \tag{4.10}$$

Eq. (3.7) shows that the disturbed uncorrelated representation vector $\hat{\mathbf{f}}_m^u$ satisfies ϵ-

differential privacy. □

Secondly, we use *Correlated Differential Privacy Mechanism* Liu et al. (2016) to achieve the correlated representations' ϵ-differential privacy by adding Laplace noise. In this paper, we use the non-negative cosine distance $Cos(\cdot,\cdot) \in [0,1]$ to measure the correlation among representations, where a higher cosine distance value means a larger correlation, and a lower cosine distance value indicates a smaller correlation. Then, we can compute the perturbed correlated representation $\hat{\mathbf{f}}_m^c$ as Eq. (4.11).

$$\hat{\mathbf{f}}_m^c = \mathbf{f}_m^c + Lap\left(0, \sum_{m' \in \{v,a,l\}} Cos(\mathbf{f}_m^c, \mathbf{f}_{m'}^c) S_{\mathbf{f}_m^c}/\epsilon\right), \tag{4.11}$$

where $S_{\mathbf{f}_m^c}$ is the global sensitivity of the uncorrelated representation vector $\mathbf{f}_m^c$ and is equal to the difference between the maximal and the minimal items in $\mathbf{f}_m^c$, and $Cos(\mathbf{f}_m^c, \mathbf{f}_{m'}^c)$ is used as the correlation coefficient between $\mathbf{f}_m^c$ and $\mathbf{f}_{m'}^c$.

Theorem 5. *By adding the Laplace noise* $Lap\left(0, \sum_{m' \in \{v,a,l\}} Cos(\mathbf{f}_m^c, \mathbf{f}_{m'}^c) S_{\mathbf{f}_m^c}/\epsilon\right)$ *into the correlated representation vector* $\mathbf{f}_m^c$*, the output perturbed correlated representation vector* $\hat{\mathbf{f}}_m^c$ *meets* ϵ*-differential privacy.*

Proof. In accordance with Liu et al. (2016), we define $QS_{\mathbf{f}_m^c} = \sum_{m' \in \{v,a,l\}} Cos(\mathbf{f}_m^c, \mathbf{f}_{m'}^c) S_{\mathbf{f}_m^c}$ as the correlated global sensitivity of the correlated representation vector $\mathbf{f}_m^c$. Similar to the proof of Theorem 1, let $\Pr[\cdot]$ be the Laplace distribution. Accordingly, there is

$$\begin{aligned} \ln \frac{\Pr[\mathbf{f}_m^c]}{\Pr[\hat{\mathbf{f}}_m^c]} &= \ln \frac{\frac{\epsilon}{2QS_{\mathbf{f}_m^c}} e^{-\frac{\epsilon}{QS_{\mathbf{f}_m^c}}|\mathbf{f}_m^c|}}{\frac{\epsilon}{2QS_{\mathbf{f}_m^c}} e^{-\frac{\epsilon}{QS_{\mathbf{f}_m^c}}|\hat{\mathbf{f}}_m^c|}} \\ &= \frac{\epsilon}{QS_{\mathbf{f}_m^c}} (|\hat{\mathbf{f}}_m^c| - |\mathbf{f}_m^c|) \le \epsilon. \end{aligned} \tag{4.12}$$

Eq. (4.12) indicates that the perturbed correlated representation vector $\hat{\mathbf{f}}_m^c$ meets ϵ-differential privacy. □

Notably, for $\hat{\mathbf{f}}_m^c$, the added Laplace noise can be lower if the value of $Cos(\mathbf{f}_m^c, \mathbf{f}_{m'}^c)$ is decreased, which can mitigate the side-effect of the Laplace noise on the sentiment prediction performance. On the other hand, as shown in $\mathcal{L}_{enc_2}$, the correlation between $\mathbf{f}_m^c$ and $\mathbf{f}_{m'}^c$ can be adjusted by changing the value of c in our correlated representation learning process, which makes the generation of privacy-preserving representations more flexible.

4.3.5 Privacy-Preserving Sentiment Prediction

Following the fusion idea of Hazarika et al. (2020), the outputs of the aforementioned differential privacy protection scheme, including $\hat{\mathbf{f}}_v^c$, $\hat{\mathbf{f}}_a^c$, $\hat{\mathbf{f}}_l^c$, $\hat{\mathbf{f}}_v^u$, $\hat{\mathbf{f}}_a^u$, and $\hat{\mathbf{f}}_l^u$, are fused into a joint vector $\hat{\mathbf{f}}_{out} \in \mathbb{R}^{d_{out}}$ through simple concatenation. Then, the prediction function G is applied to the privacy-preserving prediction task with $\hat{\mathbf{f}}_{out}$ as the input:

$$\hat{\mathbf{y}} = G(\hat{\mathbf{f}}_{out}; \theta_{out}), \tag{4.13}$$

where $\hat{\mathbf{y}}$ is the predicted label vector corresponding to $\hat{\mathbf{f}}_{out}$, and θ_{out} denotes the parameters of the prediction function.

We use *cross-entropy loss* to calculate the loss of the privacy-preserving sentiment prediction task in Eq. (4.14).

$$\mathcal{L}_{task} = -\frac{1}{n}\sum_{i=0}^{n} \mathbf{y}_i \cdot \log(\hat{\mathbf{y}}_i), \tag{4.14}$$

in which $\mathcal{L}_{task}$ is the prediction loss, n represents the number of utterances in a training batch, $\mathbf{y}_i$ is the i-th ground-truth label and $\hat{\mathbf{y}}_i$ is the i-th predicted label.

Consequently, to learn the privacy-preserving correlated and uncorrelated multimodal representations for the privacy-preserving multimodal sentiment analysis, the overall loss function of DPCRL, $\mathcal{L}_{DPCRL}$, should consist of the encoding loss $\mathcal{L}_{enc}$ in Eq. (4.6), the decoding loss $\mathcal{L}_{dec}$ in Eq. (4.7), and the privacy-preserving prediction loss $\mathcal{L}_{task}$ in Eq. (4.14) as formulated by Eq. (4.15).

$$\mathcal{L}_{DPCRL} = \alpha\mathcal{L}_{enc} + \beta\mathcal{L}_{dec} + \gamma\mathcal{L}_{task}, \tag{4.15}$$

where $\alpha, \beta, \gamma \in (0, 1]$ are the weights of the loss functions. Our DPCRL model can be learnt by minimizing $\mathcal{L}_{DPCRL}$. The specific network architectures of the encoders, E_m^c and E_m^u, the decoder D, and the prediction function G used in the DPCRL model are described in Section 4.4.1.4.

4.4 Experiment and Analysis

In this section, we first introduce our experiment settings and then present comprehensive experimental results to validate the superiority of our proposed DPCRL model over the state of the art for privacy-preserving multimodal sentiment analysis. The codes of our model and all experimental results in this paper can be found at `https://github.com/ahahnut/Differential-Private-Correlated-Representation-Learning`.

4.4.1 Experimental Settings

The datasets, baselines, performance metrics, network architectures, and hyper-parameter settings are described below.

4.4.1.1 Datasets

We use two benchmark datasets in our experiments for multimodal sentiment analysis. **CMU-MOSI (MOSI) dataset** Zadeh et al. (2016) is a collection of YouTube monologues consisting of 2198 subjective video segments (utterances), where speakers express their opinions on topics such as movies. Each utterance is manually annotated with an integer opinion score in $[-3, 3]$, where -3 and 3 represent the strongest negative and the strongest positive sentiments, respectively. **CMU-MOSEI (MOSEI) dataset** Zadeh et al. (2018c) contains 23453 annotated video segments and is an improvement of MOSI with a larger number of utterances and a greater variety in samples, speakers, and topics.

4.4.1.2 Baseline

MISA Hazarika et al. (2020), Self-MM Yu et al. (2021) and MMIM Han et al. (2021) are the currently pioneering models on both MOSI and MOSEI datasets for multimodal sentiment analysis. MISA with Differential Privacy (MISA-DP) is a simple combination of the differential privacy mechanism and MISA to obtain differentially private representations for sentiment prediction while guaranteeing privacy protection. MISA, Self-MM, MMIM, and MISA-DP are adopted as baseline mechanisms for performance comparison.

4.4.1.3 Performance Metrics

The task of sentiment prediction on MOSI and MOSEI can be treated as a classification process and evaluated via integer classification scores in $[-3, 3]$ that are so-called seven-class accuracy (Acc-7) Zadeh et al. (2016). Besides, two approaches of computing binary accuracy (Acc-2) can be also adopted to measure the performance of sentiment prediction. The first one is *Negative/Non-negative (Neg/Non-neg)* classification, where the non-negative labels are indicated by non-negative classification scores Zadeh et al. (2018b). The second one is calculated based on *Negative/Positive (Neg/Pos)* classes, where the negative and the positive classes are indicated by the negative and the positive scores, respectively Tsai et al. (2019). To sum up, Acc-2 (Neg/Non-neg), F1 (Neg/Non-neg), Acc-2 (Neg/Pos), F1 (Neg/Pos), and Acc-7 are used as performance metrics in our experiments.

Table 4.1 Ablation Study of Correlated Representation Learning Scheme on MOSI Dataset

Uncorrelated Representations	Correlated Representations	Acc-7
✓	✗	0.3145
✗	✓	0.3474
✓	✓	0.446

Table 4.2 Ablation Study of Correlated Representation Learning Scheme on MOSEI Dataset

Uncorrelated Representations	Correlated Representations	Acc-7
✓	✗	0.4056
✗	✓	0.4362
✓	✓	0.539

Table 4.3 Evaluation Results of Correlated Representation Learning Scheme on MOSI Dataset

Model	Expected Data Correlation	Trained Data Correlation
MISA Hazarika et al. (2020)	/	/
Self-MM Yu et al. (2021)	/	/
MMIM Han et al. (2021)	/	/
CRL	$c = 0.0$, $d_c = 90.00°$	$e = 0.0003$, $d_e = 89.82°$
CRL	$c = 0.1$, $d_c = 84.26°$	$e = 0.1183$, $d_e = 83.21°$
CRL	$c = 0.2$, $d_c = 78.46°$	$e = 0.2093$, $d_e = 77.92°$
CRL	$c = 0.3$, $d_c = 72.54°$	$e = 0.3069$, $d_e = 72.13°$
CRL	$c = 0.4$, $d_c = 66.42°$	$e = 0.4050$, $d_e = 66.11°$
CRL	$c = 0.5$, $d_c = 60.00°$	$e = 0.5036$, $d_e = 59.76°$
CRL	$c = 0.6$, $d_c = 53.13°$	$e = 0.6035$, $d_e = 52.88°$
CRL	$c = 0.7$, $d_c = 45.57°$	$e = 0.7029$, $d_e = 45.34°$
CRL	$c = 0.8$, $d_c = 36.87°$	$e = 0.8029$, $d_e = 36.59°$
CRL	$c = 0.9$, $d_c = 25.84°$	$e = 0.9023$, $d_e = 25.54°$
CRL	$c = 1.0$, $d_c = 0.00°$	$e = 0.9997$, $d_e = 1.40°$

Table 4.4 Evaluation Results of Correlated Representation Learning Scheme on MOSEI Dataset

Model	Expected Data Correlation	Trained Data Correlation
MISA Hazarika et al. (2020)	/	/
Self-MM Yu et al. (2021)	/	/
MMIM Han et al. (2021)	/	/
CRL	$c = 0.0$, $d_c = 90.00°$	$e = 0.0007$, $d_e = 89.60°$
CRL	$c = 0.1$, $d_c = 84.26°$	$e = 0.1034$, $d_e = 84.07°$
CRL	$c = 0.2$, $d_c = 78.46°$	$e = 0.2017$, $d_e = 78.36°$
CRL	$c = 0.3$, $d_c = 72.54°$	$e = 0.3066$, $d_e = 72.15°$
CRL	$c = 0.4$, $d_c = 66.42°$	$e = 0.4052$, $d_e = 66.10°$
CRL	$c = 0.5$, $d_c = 60.00°$	$e = 0.5040$, $d_e = 59.74°$
CRL	$c = 0.6$, $d_c = 53.13°$	$e = 0.6034$, $d_e = 52.89°$
CRL	$c = 0.7$, $d_c = 45.57°$	$e = 0.7005$, $d_e = 45.53°$
CRL	$c = 0.8$, $d_c = 36.87°$	$e = 0.8004$, $d_e = 36.83°$
CRL	$c = 0.9$, $d_c = 25.84°$	$e = 0.9022$, $d_e = 25.55°$
CRL	$c = 1.0$, $d_c = 0.00°$	$e = 0.9996$, $d_e = 1.62°$

4.4.1.4 Neural Network Architectures

In our proposed DPCRL model, the neural network architectures of the feature extraction, encoding, decoding, and sentiment prediction modules are described below. (i) **Feature Extraction.** Facial Action Coding System (FACS) Rosenberg & Ekman (2020) is applied to extract facial expression features that include facial action units and face pose. An acoustic analysis framework (COVAREP) Degottex et al. (2014) is employed to extract the acoustic features that contain 12 Mel-frequency cepstral coefficients, pitch, voiced/unvoiced segmenting features, glottal source parameters, and other features related to emotions and the tone of speech. The pre-trained BERT Devlin et al. (2018) is utilized as the feature extractor for textual utterance. Accordingly, the visual feature dimension is $d_v = 47$, the acoustic feature dimension is $d_a = 74$, and the textual feature dimension is $d_l = 784$. Furthermore, in order to align the multimodal features for our encoding process, we exploit *one Fully-Connected Layer with ReLU activation function and one Normalization Layer* to embed these features into a space with the same dimension. (ii) **Encoding.** The correlated multimodal representation encoder E^c_m is built by using *one Fully-Connected Layer with Sigmoid activation function* to extract the correlated representations. The uncorrelated multimodal representation encoder E^u_m is designed through *one Fully-Connected Layer with Sigmoid activation function* to extract the uncorrelated representations. To be specific, there are three encoders to learn the correlated representations and three encoders to learn the uncorrelated representations. Although these encoders have the same structure, their parameters are updated differently during training process to learn correlated and uncorrelated representations. (iii) **Decoding.** The decoder D is established as *one Fully-Connected Layer* for reconstruction to avoid learning unrepresentative vector of data in the encoding process. (iv) **Sentiment Prediction.** In the prediction function G, *one Transformer Encoder Layer* is used for transformation, *one Fully-Connected Layer with a Dropout Layer plus a ReLU activation function* is used for fusion, and *one Fully-Connected Layer* is used to map all representations into one dimension for final prediction.

4.4.1.5 Hyperparameter Settings

Our experiments are conducted on Ubuntu OS with a Nvidia Tesla V100 GPU and 16 GB RAM. The batch size of samples for training MOSI and MOSEI datasets are 64 and 16, respectively. The learning rate of training is set as 10^{-4}. The probabilities of dropout in the dropout layer for training MOSI and MOSEI datasets are 0.5 and 0.1, respectively. Via comprehensive ablation study, the weights of loss functions are set as $\alpha = 0.45$, $\beta = 0.1$, and $\gamma = 0.45$ for training MOSI dataset with 500 epochs, and the weights of loss functions are set to be $\alpha = 0.35$, $\beta = 0.3$, $\gamma = 0.35$ for training MOSEI dataset with 500 epochs. Besides, we vary the correlation factor c from 0 to 1 with the step of 0.1 to illustrate the effectiveness of our correlated representation learning model and set the privacy budget $\epsilon \in \{1.0, 1.5, 2.0, 2.5, 3.0\}$ to evaluate our DPCRL model.

4.4.2 Evaluation on Correlated Representation Learning (CRL)

We first present ablation study of our correlated representation learning model trained with the correlation factor $c = 0.5$ and the default hyperparameter settings. In Table 4.1 and Table 4.2, we show the results of ablation study on MOSI dataset and MOSEI dataset, respectively. By comparing these results, it is clear that the incorporation of the correlated and uncorrelated multimodal representations can obtain the best performance of the multimodal sentiment analysis, which verifies the effectiveness of our model design.

Then, we train our scheme by changing the correlation factor c from 0 to 1 with the step

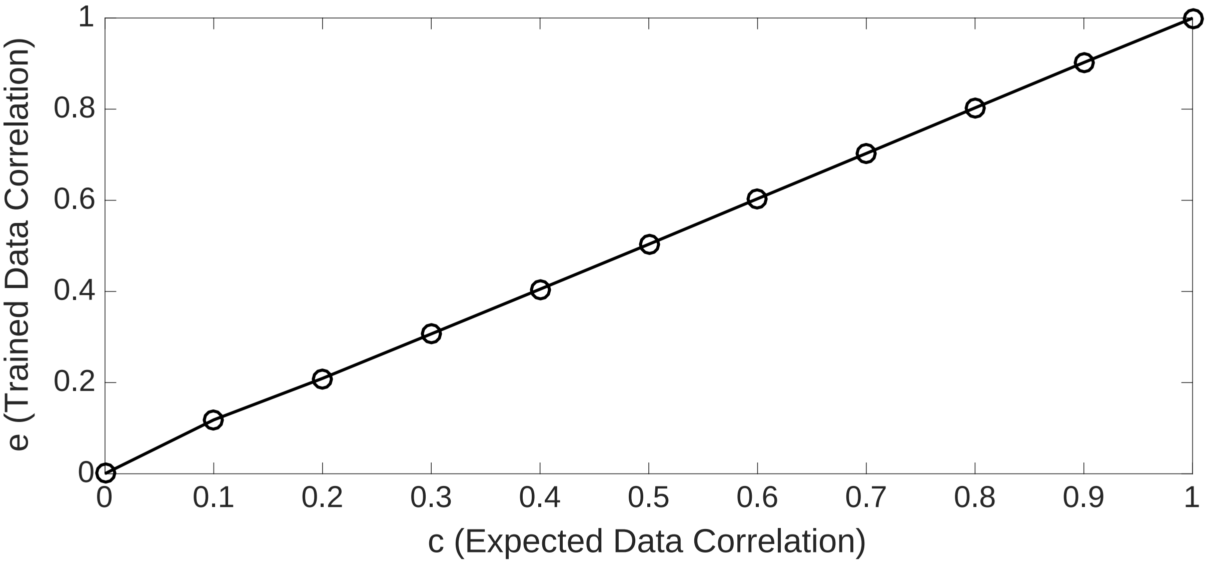

Figure 4.2 Trained Data Correlation in MOSI Dataset

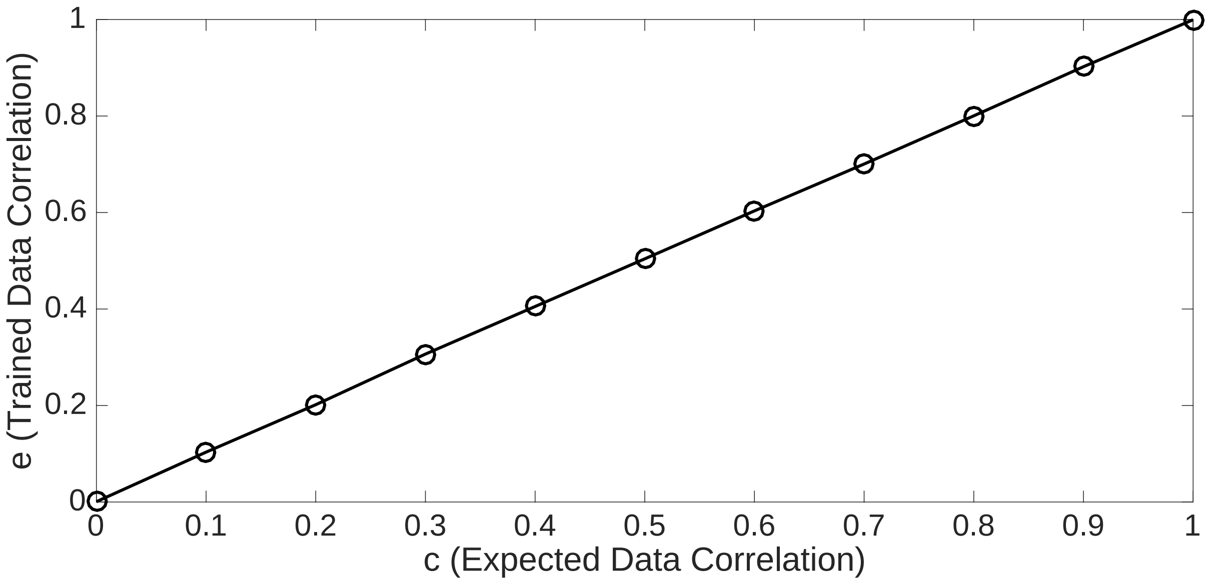

Figure 4.3 Trained Data Correlation in MOSEI Dataset

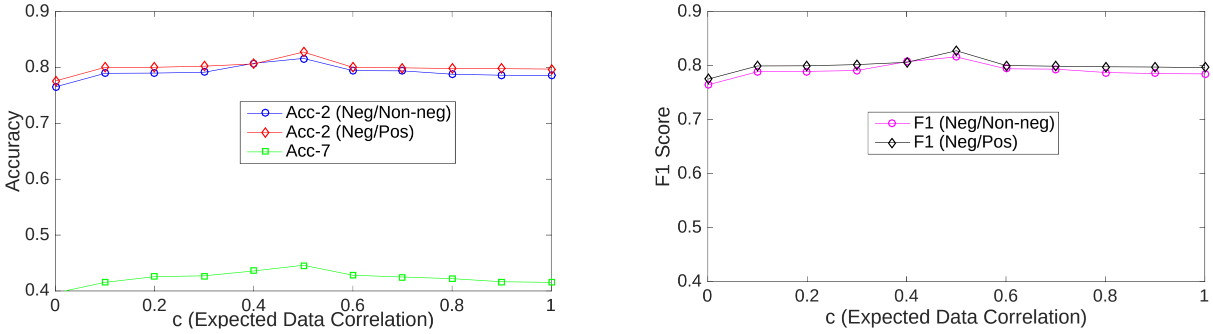

Figure 4.4 Prediction Results of CRL on MOSI Dataset

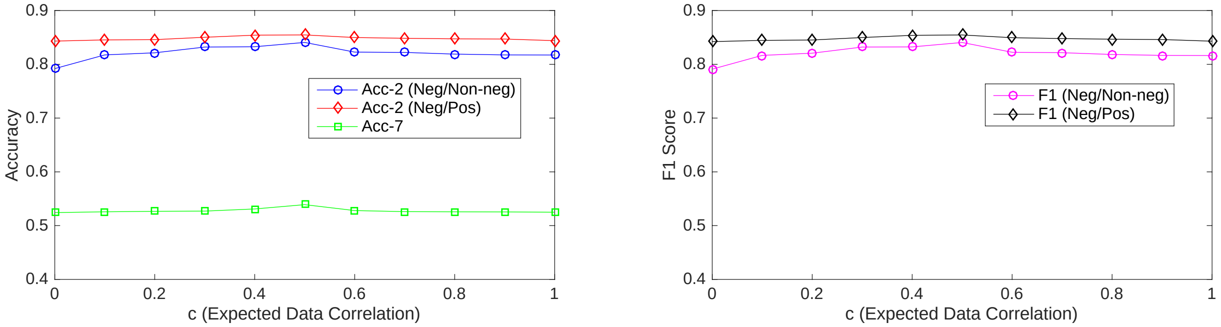

Figure 4.5 Prediction Results of CRL on MOSEI Dataset

of 0.1 to validate that c can help achieve effective heterogeneous multimodal data transformation satisfying the requirements for multimodal sentiment analysis. When the training process terminates, the correlation coefficient among the trained correlated representations is denoted by e. Since $c, e \in [0, 1]$ are the cosine values, we can calculate the angle degree, d_c, corresponding to c and the angle degree, d_e, corresponding to e. That is, c and d_c imply our expected data correlation, and e and d_e are our trained data correlation. The difference between our expected and trained data correlation can reflect the effectiveness of our proposed correlated representation learning scheme. To clearly investigate the impact of c on the performance of sentiment prediction, we compute Acc-2 (Neg/Non-neg), F1 (Neg/Non-neg), Acc-2 (Neg/Pos), F1 (Neg/Pos) and Acc-7 on the learned correlated and uncorrelated representations.

Table 4.3 presents the values of c, d_c, e, and d_e when the correlated representation learning scheme is implemented on MOSI dataset. By comparing these values, one can see that the expected data correlation is very close to the corresponding trained data correlation. For examples, $e = 0.1183$ when $c = 0.1$, and $e = 0.2093$ when $c = 0.2$. For a more explicit comparison, we plot Fig. 4.2 to examine the impact of c on e, from which we can also observe that e is nearly equal to c. The results of Table 4.3 and Fig. 4.2 confirm that in our correlated representation learning scheme, the utilization of c is effective to accomplish our expected heterogeneous multimodal data transformation. When implementing our correlated representation learning scheme on MOSEI dataset, we can obtain the same conclusion through Table 4.4 and Fig. 4.3.

Table 4.5 Evaluation Results of Acc-2 (Neg/Non-neg) on MOSI Dataset (DPCRL v.s. Baselines)

Model	$\epsilon = 1.0$	$\epsilon = 1.5$	$\epsilon = 2.0$	$\epsilon = 2.5$	$\epsilon = 3.0$
MISA Hazarika et al. (2020)	0.7857	0.7857	0.7857	0.7857	0.7857
Self-MM Yu et al. (2021)	0.783	0.783	0.783	0.783	0.783
MMIM Han et al. (2021)	0.799	0.799	0.799	0.799	0.799
MISA-DP	0.4533	0.4543	0.4606	0.4664	0.4766
DPCRL ($c = 0.1$)	0.7842	0.7857	0.7789	0.8002	0.7725
DPCRL ($c = 0.2$)	0.7886	0.774	0.7798	0.7827	0.8002
DPCRL ($c = 0.3$)	0.7988	0.7944	0.7914	0.7944	0.7711
DPCRL ($c = 0.4$)	0.7784	0.7827	0.7609	0.7842	0.8032
DPCRL ($c = 0.5$)	0.7653	0.7784	0.7784	0.7827	0.7769

The multimodal representations learned from our correlated representation learning scheme

Table 4.6 Evaluation Results of F1 (Neg/Non-neg) on MOSI Dataset (DPCRL v.s. Baselines)

Model	$\epsilon = 1.0$	$\epsilon = 1.5$	$\epsilon = 2.0$	$\epsilon = 2.5$	$\epsilon = 3.0$
MISA Hazarika et al. (2020)	0.7847	0.7847	0.7847	0.7847	0.7847
Self-MM Yu et al. (2021)	0.7834	0.7834	0.7834	0.7834	0.7834
MMIM Han et al. (2021)	0.7984	0.7984	0.7984	0.7984	0.7984
MISA-DP	0.421	0.422	0.4283	0.4341	0.4443
DPCRL ($c = 0.1$)	0.7832	0.7848	0.7793	0.7993	0.7715
DPCRL ($c = 0.2$)	0.7881	0.7733	0.7793	0.782	0.7997
DPCRL ($c = 0.3$)	0.7985	0.7943	0.7913	0.7943	0.7708
DPCRL ($c = 0.4$)	0.7779	0.7772	0.7606	0.7835	0.8029
DPCRL ($c = 0.5$)	0.7643	0.7771	0.7779	0.7818	0.7759

Table 4.7 Evaluation Results of Acc-2 (Neg/Pos) on MOSI Dataset (DPCRL v.s. Baselines)

Model	$\epsilon = 1.0$	$\epsilon = 1.5$	$\epsilon = 2.0$	$\epsilon = 2.5$	$\epsilon = 3.0$
MISA Hazarika et al. (2020)	0.7972	0.7972	0.7972	0.7972	0.7972
Self-MM Yu et al. (2021)	0.8079	0.8079	0.8079	0.8079	0.8079
MMIM Han et al. (2021)	0.8208	0.8208	0.8208	0.8208	0.8207
MISA-DP	0.4298	0.4329	0.439	0.4496	0.4573
DPCRL ($c = 0.1$)	0.8018	0.7942	0.8048	0.8201	0.7911
DPCRL ($c = 0.2$)	0.7978	0.7926	0.7911	0.7942	0.8109
DPCRL ($c = 0.3$)	0.814	0.8033	0.8033	0.814	0.7835
DPCRL ($c = 0.4$)	0.782	0.7987	0.7698	0.7926	0.8201
DPCRL ($c = 0.5$)	0.7743	0.7896	0.7911	0.7972	0.7881

are exploited to evaluate the performance of sentiment analysis in terms of Acc-2 (Neg/Non-neg), F1 (Neg/Non-neg), Acc-2 (Neg/Pos), F1 (Neg/Pos) and Acc-7. These experimental results on MOSI dataset are presented in Table 4.3. Take the values of Acc-2 (Neg/Non-neg) as an example for analysis: (i) The values of Acc-2 (Neg/Non-neg) obtained via MISA, Self-MM, and MMIM are 0.7857, 0.783, and 0.799, respectively. While, the value of Acc-2 (Neg/Non-neg) obtained in our correlated representation learning scheme falls in $[0.7653, 0.8163]$ when

Table 4.8 Evaluation Results of F1 (Neg/Pos) on MOSI Dataset (DPCRL v.s. Baselines)

Model	$\epsilon = 1.0$	$\epsilon = 1.5$	$\epsilon = 2.0$	$\epsilon = 2.5$	$\epsilon = 3.0$
MISA Hazarika et al. (2020)	0.8092	0.8092	0.8092	0.8092	0.8092
Self-MM Yu et al. (2021)	0.8066	0.8066	0.8066	0.8066	0.8066
MMIM Han et al. (2021)	0.8173	0.8173	0.8173	0.8173	0.8173
MISA-DP	0.403	0.4061	0.4122	0.4228	0.4305
DPCRL ($c = 0.1$)	0.8009	0.7937	0.8039	0.8191	0.7902
DPCRL ($c = 0.2$)	0.7971	0.7923	0.7904	0.7937	0.8102
DPCRL ($c = 0.3$)	0.8139	0.8032	0.8032	0.8137	0.7834
DPCRL ($c = 0.4$)	0.7819	0.7984	0.7691	0.7921	0.8194
DPCRL ($c = 0.5$)	0.7734	0.7891	0.7902	0.7962	0.7872

Table 4.9 Evaluation Results of Acc-7 on MOSI Dataset (DPCRL v.s. Baselines)

Model	$\epsilon = 1.0$	$\epsilon = 1.5$	$\epsilon = 2.0$	$\epsilon = 2.5$	$\epsilon = 3.0$
MISA Hazarika et al. (2020)	0.4154	0.4154	0.4154	0.4154	0.4154
Self-MM Yu et al. (2021)	0.4244	0.4244	0.4244	0.4244	0.4244
MMIM Han et al. (2021)	0.433	0.433	0.433	0.433	0.433
MISA-DP	0.1529	0.1545	0.156	0.1574	0.159
DPCRL ($c = 0.1$)	0.3892	0.3877	0.4081	0.4096	0.395
DPCRL ($c = 0.2$)	0.3979	0.3979	0.3862	0.3848	0.411
DPCRL ($c = 0.3$)	0.4227	0.4189	0.4139	0.4154	0.4285
DPCRL ($c = 0.4$)	0.3877	0.3833	0.3862	0.4052	0.4387
DPCRL ($c = 0.5$)	0.3615	0.379	0.4081	0.4037	0.395

the value of c varies from 0 to 1 with the step of 0.1. Especially, when $c = 0.5$ (*i.e.*, the angle degree is $d_c = 60°$), the value of Acc-2 (Neg/Non-neg) reaches 0.8163. Thus, we can conclude that our correlated representation learning scheme and the baselines (including MISA, Self-MM, and MMIM) have comparable performance in terms of Acc-2 (Neg/Non-neg). (ii) For our correlated representation learning scheme, the value of Acc-2 (Neg/Non-neg) increases with the growth of c when $c \in [0.0, 0.5]$, which indicates that the increased similarity among

Table 4.10 Evaluation Results of Acc-2 (Neg/Non-neg) on MOSEI Dataset (DPCRL v.s. Baselines)

Model	$\epsilon = 1.0$	$\epsilon = 1.5$	$\epsilon = 2.0$	$\epsilon = 2.5$	$\epsilon = 3.0$
MISA Hazarika et al. (2020)	0.8173	0.8173	0.8173	0.8173	0.8173
Self-MM Yu et al. (2021)	0.7944	0.7944	0.7944	0.7944	0.7944
MMIM Han et al. (2021)	0.79	0.79	0.79	0.79	0.79
MISA-DP	0.6945	0.697	0.706	0.8003	0.8044
DPCRL ($c = 0.1$)	0.8177	0.8231	0.8102	0.8061	0.8096
DPCRL ($c = 0.2$)	0.8289	0.8167	0.8083	0.8061	0.8171
DPCRL ($c = 0.3$)	0.8336	0.8169	0.8113	0.8137	0.8122
DPCRL ($c = 0.4$)	0.8263	0.8227	0.801	0.8072	0.8098
DPCRL ($c = 0.5$)	0.8242	0.8186	0.8083	0.8117	0.8098

Table 4.11 Evaluation Results of F1 (Neg/Non-neg) on MOSEI Dataset (DPCRL v.s. Baselines)

Model	$\epsilon = 1.0$	$\epsilon = 1.5$	$\epsilon = 2.0$	$\epsilon = 2.5$	$\epsilon = 3.0$
MISA Hazarika et al. (2020)	0.8193	0.8193	0.8193	0.8193	0.8193
Self-MM Yu et al. (2021)	0.7995	0.7995	0.7995	0.7995	0.7995
MMIM Han et al. (2021)	0.7966	0.7966	0.7966	0.7966	0.7966
MISA-DP	0.6758	0.6783	0.6873	0.7874	0.7911
DPCRL ($c = 0.1$)	0.8172	0.8222	0.8092	0.8052	0.8091
DPCRL ($c = 0.2$)	0.8286	0.816	0.8078	0.8052	0.8168
DPCRL ($c = 0.3$)	0.8335	0.8168	0.811	0.8136	0.8121
DPCRL ($c = 0.4$)	0.826	0.822	0.8005	0.8065	0.8095
DPCRL ($c = 0.5$)	0.8237	0.8177	0.8073	0.8108	0.8095

representations is helpful to improve the performance of sentiment prediction. (iii) In our correlated representation learning scheme, the value of Acc-2 (Neg/Non-neg) gradually decreases with the growth of c when $c \in [0.6, 1.0]$, which implies that the decreased diversity among representations degrades the performance of sentiment prediction. (iv) The correlation factor c can be used to balance the trade-off between representation similarity and

Table 4.12 Evaluation Results of Acc-2 (Neg/Pos) on MOSEI Dataset (DPCRL v.s. Baselines)

Model	$\epsilon = 1.0$	$\epsilon = 1.5$	$\epsilon = 2.0$	$\epsilon = 2.5$	$\epsilon = 3.0$
MISA Hazarika et al. (2020)	0.844	0.844	0.844	0.844	0.844
Self-MM Yu et al. (2021)	0.8122	0.8122	0.8122	0.8122	0.8122
MMIM Han et al. (2021)	0.8223	0.8223	0.8223	0.8223	0.8223
MISA-DP	0.6228	0.6244	0.6288	0.8316	0.8385
DPCRL ($c = 0.1$)	0.85	0.8542	0.8492	0.8487	0.8396
DPCRL ($c = 0.2$)	0.8569	0.8531	0.8401	0.8506	0.8545
DPCRL ($c = 0.3$)	0.8536	0.8523	0.8371	0.8545	0.8506
DPCRL ($c = 0.4$)	0.8528	0.855	0.8476	0.8473	0.8492
DPCRL ($c = 0.5$)	0.8517	0.8518	0.8484	0.8545	0.8476

Table 4.13 Evaluation Results of F1 (Neg/Pos) on MOSEI Dataset (DPCRL v.s. Baselines)

Model	$\epsilon = 1.0$	$\epsilon = 1.5$	$\epsilon = 2.0$	$\epsilon = 2.5$	$\epsilon = 3.0$
MISA Hazarika et al. (2020)	0.842	0.842	0.842	0.842	0.842
Self-MM Yu et al. (2021)	0.825	0.825	0.825	0.825	0.825
MMIM Han et al. (2021)	0.8351	0.8351	0.8351	0.8351	0.8351
MISA-DP	0.6118	0.6134	0.6158	0.8206	0.8275
DPCRL ($c = 0.1$)	0.8491	0.8532	0.8483	0.8482	0.8387
DPCRL ($c = 0.2$)	0.8562	0.8526	0.8394	0.8503	0.8538
DPCRL ($c = 0.3$)	0.8535	0.852	0.837	0.8544	0.8505
DPCRL ($c = 0.4$)	0.8521	0.8545	0.8469	0.847	0.8485
DPCRL ($c = 0.5$)	0.8508	0.8507	0.8475	0.854	0.8467

representation diversity for improving multimodal sentiment analysis performance.

Similarly, by analyzing the results of F1 (Neg/Non-neg), Acc-2 (Neg/Pos), F1 (Neg/Pos), and Acc-7 on MOSI dataset in Table 4.3, we can draw the same conclusions. In order to explicitly show the impact of c on sentiment prediction, we present the results of Acc-2 (Neg/Non-neg), F1 (Neg/Non-neg), Acc-2 (Neg/Pos), F1 (Neg/Pos) and Acc-7 on MOSI dataset in Fig. 4.4 for comparison. Moreover, as shown in Table 4.4 and Fig. 4.5, the

Table 4.14 Evaluation Results of Acc-7 on MOSEI Dataset (DPCRL v.s. Baselines)

Model	$\epsilon = 1.0$	$\epsilon = 1.5$	$\epsilon = 2.0$	$\epsilon = 2.5$	$\epsilon = 3.0$
MISA Hazarika et al. (2020)	0.5249	0.5249	0.5249	0.5249	0.5249
Self-MM Yu et al. (2021)	0.5159	0.5159	0.5159	0.5159	0.5159
MMIM Han et al. (2021)	0.5237	0.5237	0.5237	0.5237	0.5237
MISA-DP	0.4042	0.4101	0.4142	0.4379	0.5025
DPCRL ($c = 0.1$)	0.5109	0.5077	0.5197	0.5182	0.5242
DPCRL ($c = 0.2$)	0.5098	0.5116	0.5088	0.5217	0.5182
DPCRL ($c = 0.3$)	0.5083	0.507	0.5133	0.5163	0.5206
DPCRL ($c = 0.4$)	0.5084	0.5128	0.5129	0.5131	0.5193
DPCRL ($c = 0.5$)	0.5131	0.5083	0.5186	0.515	0.5199

experimental results on MOSEI dataset can also confirm our aforementioned analysis.

4.4.3 Evaluation on Our DPCRL Model

In our proposed DPCRL model, there are two system parameters, ϵ and c. The value of ϵ, which is so-called "privacy budget", indicates the degree of privacy protection. A smaller ϵ implies a higher degree of data privacy protection. We implement our DPCRL model with $\epsilon = 1.0, 1.5, 2.0, 2.5, 3.0$ on datasets, which is reasonable and applicable in real applications for privacy protection based on the differential privacy mechanisms. The value of c represents the expected correlation among the learned correlated representations. A larger c implies a closer correlation among the correlated representations. In our experiments, we set $c = 0.1, 0.2, 0.3, 0.4, 0.5$ with the following considerations. (i) From Table 4.3 and Table 4.4, the prediction performance of our correlated representation learning scheme with $c = 0.0$ is worse than that of the state of the art (MISA). Therefore, it may not be suitable to set $c = 0.0$ when we aim to maintain prediction performance as much as possible while ensuring differential privacy protection. (ii) We attempt to learn the correlated representations with a relatively lower value of c so as to decrease the side-effect of the additional Laplace noise on prediction performance.

In Table 4.5, we compare the Acc-2 (Neg/Non-neg) results of our DPCRL model and the two baseline models on MOSI dataset. We take Acc-2 (Neg/Non-neg) of DPCRL with $c = 0.1$ as an example to illustrate the effectiveness of our proposed DPCRL model: (i) By comparing the Acc-2 (Neg/Non-neg) values, it can be found that the performance of DPCRL is comparable to that of baselines (including MISA, Self-MM, and MMIM), which indicates that our DPCRL model can maintain the performance of sentiment analysis while satisfying differential privacy guarantee. (ii) By comparing Acc-2 (Neg/Non-neg) values of MISA-DP and DPCRL with a same value of ϵ, we can see that the Acc-2 (Neg/Non-neg) values of our proposed DPCRL model are much higher than those of the baseline model MISA-DP which uses the invariant data representations with the correlation $c = 1.0$. That is, with the same privacy budget ϵ, our DPCRL model outperforms MISA-DP from the aspect of maintaining the sentiment prediction performance. The main reason is that our correlated representation learning scheme used in DPCRL can be leveraged to learn the correlated representations with a relatively lower correlation factor, mitigating the side-effect of the additional Laplace noise on the sentiment analysis.

For a comprehensive demonstration, we present F1 (Neg/Non-neg), Acc-2 (Neg/Pos), F1 (Neg/Pos) and Acc-7 of our DPCRL model and the baselines on MOSI dataset in Table 4.6, Table 4.7, Table 4.8 and Table 4.9, respectively. Additionally, for MOSEI dataset, the values of Acc-2 (Neg/Non-neg), F1 (Neg/Non-neg), Acc-2 (Neg/Pos), F1 (Neg/Pos), and Acc-7 of our DPCRL model and baselines are presented in Table 4.10-Table 4.14.

Based on the above analysis, we obtain the following critical conclusions: (i) Our proposed DPCRL model is effective to accomplish privacy-preserving multimodal sentiment analysis with providing ϵ-differential privacy guarantee. (ii) By setting a correlation factor as input, our DPCRL model can realize heterogeneous multimodal data transformation that satisfies our learning expectation. (iii) A smaller value of the correlation factor can help reduce Laplace noise added in ϵ-differential privacy mechanisms, mitigating the loss of prediction performance. (iv) Compared with the state of the art, our DPCRL model can effectively maintain and even enhance the performance of sentiment prediction while ensuring ϵ-differential privacy.

4.5 Conclusion

In this paper, we propose a DPCRL model with an aim of learning privacy-preserving data representations for multimodal sentiment analysis. Our DPCRL model consists of a novel correlated representation learning scheme and a differential privacy protection scheme. The correlated representation learning scheme can achieve heterogeneous multimodal data transformation to learn correlated and uncorrelated representations for multimodal sentiment prediction while reducing privacy leakage. The differential privacy protection scheme can produce the perturbed correlated and uncorrelated representations through inserting Laplace noise for ϵ-differential privacy. In our DPCRL model, a correlation factor is employed to learn the correlated representations for mitigating the side-effect of the additional Laplace noise on the sentiment prediction performance. Finally, the experiment results can confirm that our proposed DPCRL model outperforms the state of the art in the performance of multimodal sentiment prediction and data privacy protection.

CHAPTER 5
FUTURE WORK

5.1 Future Work 1: Multi-sensor Data Privacy Protection

Nowadays, service providers offer numerous online artificial intelligence services that utilize multi-sensor data collection. The multi-sensor data are collected from the users' device side and then the collected multiple sensor data are applied to realize the multi-sensor data prediction on the semi-honest server. Unfortunately, there is a risk of unauthorized interception of the transmitted data, potentially leading to privacy leakage. Fortunately, previous research has demonstrated the efficacy of the differential privacy mechanism for privacy protection in online AI services. Inspired by these works, we plan to propose a differential private online multi-sensor data prediction model to safeguard the privacy of these multi-sensor data prior to their transmission, which can help users avoid privacy leakage caused by attackers who can leverage the eavesdropped representations during transmission to infer the users' sensitive data via some effective deep learning attack models.

Different from the previous differential private learning models, in order to realize privacy enhanced online multi-sensor data prediction, we consider both intra-correlation and inter-correlation among multi-sensor data in the design of additional Laplace noise to ensure the fulfillment of the differential privacy guarantee. Moreover, we will define two metrics, including robustness measurement and performance bias to study the influence of additional Laplace noise on learning performance. To be specific, our objective is to address three fundamental inquiries:

- What is the impact of privacy budget on the robustness of the multi-sensor data prediction model?
- How does privacy budget affect the bias of the multi-sensor data prediction?
- What is the relationship between the model's robustness and the prediction's bias?

5.2 Future Work 2: Model Privacy Protection

Side-channel attacks pose a significant threat when the adversary gains physical access to the device, making edge-based machine-learning accelerators highly susceptible to such attacks. In light of this, it can be believed that the remote physical side-channel attacks may possibly be implemented on deep neural networks applied in real cloud-based applications. By investigating side-channel attacks on deep neural networks, we can gain valuable insights into their vulnerabilities and enhance our understanding of their susceptibility to such attacks.

As the market for edge-based deep learning hardware is projected to experience significant growth in the coming years, it becomes imperative to prioritize the development of effective and resilient side-channel defenses for deep learning applications. Unfortunately, the research on developing adequate countermeasures for side-channel attacks on deep neural networks remains relatively immature. To this end, our ideas focus on constructing more robust and efficient countermeasures to address this critical gap in the field.

Therefore, I will expand my research scope from data privacy protection to model privacy protection by investigating the side-channel attacks on deep neural networks and their corresponding countermeasures.

CHAPTER 6
CONCLUSION

This dissertation conducts research on the design of privacy-preserving deep learning mechanisms on multimedia data-oriented applications. The ideas of our proposed data privacy protection mechanisms simultaneously take into account three aspects, including specific application scenarios, privacy leakage ways, and data characteristics.

Firstly, the dissertation designs a cycle vector-quantized variational autoencoder framework to encode and decode the video with its extracted audio, which takes the advantage of multiple heterogeneous data sources in the video itself to protect individuals' privacy. The proposed framework can simultaneously achieve visual privacy protection, visual quality preservation, and video transmission efficiency.

Secondly, the dissertation proposes a differential private deep learning model to defend black-box attack and avoid large aggregated noise simultaneously by implementing differential privacy on the model's outputs. In particular, a regularization term is exploited in the loss function to increase the model prediction accuracy and robustness, and a proper bounded global sensitivity in differential privacy is designed with the intention of decreasing the bias of loss function and increasing prediction accuracy.

Thirdly, this dissertation creates a differential private correlated representation learning model to accomplish a joint consideration of data correlation and privacy protection guarantee. A correlated representation learning scheme aims to achieve heterogeneous multimodal data transformation, in which a pre-determined correlation factor is employed to flexibly adjust the expected correlation among the correlated representations. And a differential privacy protection scheme is used to obtain the disturbed correlated and uncorrelated representations by adding Laplace noise for differential privacy guarantee.

All the proposed solutions have been meticulously examined and validated through comprehensive evaluations. Additionally, within our dissertation, we delve into two prospective works for further research. In essence, our dissertation offers a comprehensive set of solutions for safeguarding multimedia data privacy in deep learning applications, with a thorough consideration of three key facets: application scenarios, potential privacy vulnerabilities, and data attributes. We are confident that the findings within this dissertation will serve as a valuable reference for enhancing multimedia data privacy protection in the context of deep learning applications.

REFERENCES

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. 2016, in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 308–318

Akhtar, M. S., Chauhan, D. S., Ghosal, D., Poria, S., Ekbal, A., & Bhattacharyya, P. 2019, CoRR, abs/1905.05812

Amin, K., Kulesza, A., Munoz, A., & Vassilvtiskii, S. 2019, in International Conference on Machine Learning, 263–271

Anderson, M. R., Cafarella, M., Ros, G., & Wenisch, T. F. 2019, in 2019 IEEE 35th International Conference on Data Engineering (ICDE), IEEE, 1466–1477

Beaulieu-Jones, B. K., Wu, Z. S., Williams, C., Lee, R., Bhavnani, S. P., Byrd, J. B., & Greene, C. S. 2019, Circulation: Cardiovascular Quality and Outcomes, 12, e005122

Bengio, Y., Courville, A., & Vincent, P. 2013, IEEE transactions on pattern analysis and machine intelligence, 35, 1798

Benini, S., Khan, K., Leonardi, R., Mauro, M., & Migliorati, P. 2019, Signal Processing: Image Communication, 74, 21

Bollen, J., Mao, H., & Zeng, X. 2011, Journal of computational science, 2, 1

Bousmalis, K., Trigeorgis, G., Silberman, N., Krishnan, D., & Erhan, D. 2016a, CoRR, abs/1608.06019

Bousmalis, K., Trigeorgis, G., Silberman, N., Krishnan, D., & Erhan, D. 2016b, in Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain, ed. D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, & R. Garnett, 343–351

Boyat, A. K., & Joshi, B. K. 2014, in 2014 IEEE International Conference on Computational Intelligence and Computing Research, IEEE, 1–6

Brkić, K., Hrkać, T., Kalafatić, Z., & Sikirić, I. 2017, IET Signal Processing, 11, 1062

Brkic, K., Sikiric, I., Hrkac, T., & Kalafatic, Z. 2017, in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 1319–1328

Cai, C., He, Y., Sun, L., Lian, Z., Liu, B., Tao, J., Xu, M., & Wang, K. 2021a, in Proceedings of the 2nd on Multimodal Sentiment Analysis Challenge, 61–67

Cai, Z., & He, Z. 2019, in 2019 IEEE 39th international conference on distributed computing systems (ICDCS), IEEE, 144–153

Cai, Z., He, Z., Guan, X., & Li, Y. 2016, IEEE Transactions on Dependable and Secure Computing, 15, 577

Cai, Z., Xiong, Z., Xu, H., Wang, P., Li, W., & Pan, Y. 2021b, ACM Computing Surveys (CSUR), 54, 1

Cai, Z., & Zheng, X. 2018, IEEE Transactions on Network Science and Engineering, 7, 766

Cai, Z., Zheng, X., Wang, J., & He, Z. 2023, IEEE Transactions on Mobile Computing, 22, 4881

Cai, Z., Zheng, X., & Yu, J. 2019, IEEE Transactions on Industrial Informatics, 15, 6492

Capital, L. 2017, Tech reports

Chamikara, M. A. P., Bertok, P., Khalil, I., Liu, D., & Camtepe, S. 2020, Computers & Security, 97, 101951

Chaudhuri, S. et al. 2018, arXiv preprint arXiv:1808.00606

Chauhan, D. S., Akhtar, M. S., Ekbal, A., & Bhattacharyya, P. 2019, in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, ACL, 5651–5661

Chen, F., & Luo, Z. 2019, CoRR, abs/1904.08138

Chen, M., Wang, S., Liang, P. P., Baltrušaitis, T., Zadeh, A., & Morency, L.-P. 2017, in Proceedings of the 19th ACM International Conference on Multimodal Interaction, ACM, 163–171

Chen, Y., Lai, Y.-K., & Liu, Y.-J. 2018, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 9465–9474

Chen, Y., Pan, Y., Yao, T., Tian, X., & Mei, T. 2019a, in Proceedings of the 27th ACM International Conference on Multimedia, 647–655

Chen, Z.-M., Wei, X.-S., Wang, P., & Guo, Y. 2019b, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5177–5186

Chi, C., Wang, Y., Tong, X., Siddula, M., & Cai, Z. 2021, IEEE Internet of Things Journal, 9, 12125

Chu, K.-Y., Kuo, Y.-H., & Hsu, W. H. 2013, in Proceedings of the 21st ACM international conference on Multimedia, 597–600

De, S., Bermudez-Edo, M., Xu, H., & Cai, Z. 2022, IEEE Transactions on Industrial Informatics, 18, 5728

Degottex, G., Kane, J., Drugman, T., Raitio, T., & Scherer, S. 2014, in 2014 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, 960–964

Devlin, J., Chang, M., Lee, K., & Toutanova, K. 2018, CoRR, abs/1810.04805

Ding, X., Fang, H., Zhang, Z., Choo, K.-K. R., & Jin, H. 2020, IEEE Transactions on Knowledge and Data Engineering

Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., & Naor, M. 2006, in Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 486–503

Eltoft, T., Kim, T., & Lee, T.-W. 2006, IEEE Signal Processing Letters, 13, 300

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. 2010, International journal of computer vision, 88, 303

Gao, R., Oh, T.-H., Grauman, K., & Torresani, L. 2019, arXiv preprint arXiv:1912.04487

Ge, W., Yang, S., & Yu, Y. 2018a, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1277–1286

Ge, Z., Mahapatra, D., Sedai, S., Garnavi, R., & Chakravorty, R. 2018b, arXiv preprint arXiv:1807.07247

George, M., & Floerkemeier, C. 2014, in European Conference on Computer Vision, Springer, 440–455

Ghanem, S., Imran, A., & Athitsos, V. 2019, in Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments, 236–242

Ghosal, D., Akhtar, M. S., Chauhan, D., Poria, S., Ekbal, A., & Bhattacharyya, P. 2018, in Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, ACL, 3454–3466

Grecos, C., & Yang, M. Y. 2005, IEEE Transactions on Broadcasting, 51, 256

Gu, C. et al. 2018a, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 6047–6056

Gu, Y., Li, X., Huang, K., Fu, S., Yang, K., Chen, S., Zhou, M., & Marsic, I. 2018b, in Proceedings of the 26th ACM international conference on Multimedia, ACM, 537–545

Hajian, S., & Domingo-Ferrer, J. 2012, in 2012 IEEE 12th International Conference on Data Mining Workshops, IEEE, 352–359

Han, W., Chen, H., & Poria, S. 2021, arXiv preprint arXiv:2109.00412

Hazarika, D., Zimmermann, R., & Poria, S. 2020, in Proceedings of the 28th ACM International Conference on Multimedia, ACM, 1122–1131

He, K., Zhang, X., Ren, S., & Sun, J. 2016, in Proceedings of the IEEE conference on computer vision and pattern recognition, 770–778

He, Z., Cai, Z., & Yu, J. 2017, IEEE Transactions on Vehicular Technology, 67, 665

He, Z., Wang, L., & Cai, Z. 2023, IEEE Internet of Things Journal

Hitaj, B., Ateniese, G., & Perez-Cruz, F. 2017, in Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 603–618

Hu, Z., & Yang, J. 2020, Plos one, 15

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. 2017, in Proceedings of the

IEEE conference on computer vision and pattern recognition, 4700–4708

Huang, Y., Li, Y. J., & Cai, Z. 2023, Big Data Mining and Analytics, 6, 234

Huang, Z., Wang, L., Shen, H. T., Shao, J., & Zhou, X. 2009, in 2009 IEEE 25th International Conference on Data Engineering, IEEE, 1511–1514

Hung, H., & Ba, S. O. 2009, Speech/non-speech detection in meetings from automatically extracted low resolution visual features, Tech. rep., Idiap

Hyvärinen, A., & Oja, E. 1997, Neural computation, 9, 1483

IBM, & the Ponemon Institute. 2019, Cost of a Data Breach Report highlights, `https://www.ibm.com/security/data-breach`

Jaiswal, M., & Provost, E. M. 2020, in Proceedings of the AAAI Conference on Artificial Intelligence, AAAI, 7985–7993

Jansen, B., Zhang, M., & Sobel, K. 2009, Journal of the American society for information science and technology, 60, 2169

Jiang, R., Qu, C., Wang, J., Wang, C., & Zheng, Y. 2020, in 2020 IEEE 36th International Conference on Data Engineering (ICDE), IEEE, 1810–1813

Jiang, W., Xie, C., & Zhang, Z. 2016in , AAAI

Kasiviswanathan, S. P., Lee, H. K., Nissim, K., Raskhodnikova, S., & Smith, A. 2011, SIAM Journal on Computing, 40, 793

Khorram, S., Jaiswal, M., Gideon, J., McInnis, M. G., & Provost, E. M. 2018, CoRR, abs/1806.10658

Kim, T., & Yang, J. 2019, IEEE Access, 7, 84992

Kipf, T. N., & Welling, M. 2017, in Proceedings of the 5th International Conference on Learning Representations

Lee, H. K., Malkin, T., & Nahum, E. 2007, in Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement, ACM, 83–92

Li, C., Wang, R., Li, J., & Fei, L. 2020a, in Recent Trends in Intelligent Computing, Communication and Devices (Springer), 277–284

Li, C., Zhong, Q., Xie, D., & Pu, S. 2019a, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 7872–7881

Li, H., He, Y., Sun, L., Cheng, X., & Yu, J. 2016a, in IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, IEEE, 1–9

Li, J. et al. 2019b, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5060–5069

Li, K., Lu, G., Luo, G., & Cai, Z. 2020b, in Proceedings of the 29th ACM International Conference on Information & Knowledge Management, ACM, 745–754

Li, K., Luo, G., Ye, Y., Li, W., Ji, S., & Cai, Z. 2020c, IEEE Internet of Things Journal, 8, 6904

Li, Y., Huang, C., Loy, C. C., & Tang, X. 2016b, in European Conference on Computer Vision, Springer, 684–700

Liang, Y., Cai, Z., Yu, J., Han, Q., & Li, Y. 2018, IEEE Network, 32, 8

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. 2014, in European conference on computer vision, Springer, 740–755

Liu, C., Chakraborty, S., & Mittal, P. 2016, in NDSS, ISOC, 21–24

Liu, S., & Kong, L. 2018, in 2018 8th International Conference on Social science and Education Research (SSER 2018), Atlantis Press

Liu, X., Lee, J.-Y., & Jin, H. 2019, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4273–4281

Liu, Z., Shen, Y., Lakshminarasimhan, V. B., Liang, P. P., Zadeh, A., & Morency, L. 2018, CoRR, abs/1806.00064

Lundmark, M., & Dahlman, C.-J. 2017, Differential privacy and machine learning: Calculating sensitivity with generated data sets

Mai, S., Hu, H., & Xing, S. 2019a, in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, ACL, 481–492

Mai, S., Hu, H., & Xing, S. 2020, in Proceedings of the AAAI Conference on Artificial Intelligence, AAAI, 164–172

Mai, S., Xing, S., & Hu, H. 2019b, IEEE Transactions on Multimedia, 22, 122

Majumder, N., Hazarika, D., Gelbukh, A., Cambria, E., & Poria, S. 2018, Knowledge-Based Systems, 161, 124

Mallya, A., Wang, T.-C., Sapra, K., & Liu, M.-Y. 2020, arXiv preprint arXiv:2007.08509

McMahan, H. B., Andrew, G., Erlingsson, U., Chien, S., Mironov, I., Papernot, N., & Kairouz, P. 2018, arXiv preprint arXiv:1812.06210

Meenpal, T., Balakrishnan, A., & Verma, A. 2019, in 2019 4th International Conference on Computing, Communications and Security (ICCCS), IEEE, 1–5

Meng, R., Cui, Q., Zhou, Z., Fu, Z., & Sun, X. 2019, IEEE Access

Mihalcea, R. 2012, WASSA 2012, 1

Mirjalili, V., Raschka, S., Namboodiri, A., & Ross, A. 2018, in 2018 International Conference on Biometrics (ICB), IEEE, 82–89

Nasir, A. F. A., Ghani, A. S. A., Zakaria, M. A., Majeed, A. P. A., & Ibrahim, A. N. 2019, Mekatronika, 1, 58

Olson, D. 1977, Harvard Educational Review, 47, 257

Ong, Y. P., Tan, C. M., & Siau, C. J. Y. 2019, Systems and methods for processing a video stream during live video sharing, uS Patent 10,412,318

Onohara, T., Ueda, R., Daini, K., Yoshio, T., Kawabe, Y., Iwayagano, S., Takuma, H., & Sakai, E. 2019, Information-sharing device, method, and terminal device for sharing application information, uS Patent 10,282,316

Ou, L., Qin, Z., Liu, Y., Yin, H., Hu, Y., & Chen, H. 2016, in 2016 IEEE 22nd International Conference on Parallel and Distributed Systems, IEEE, 422–429

Pang, J., Huang, Y., Xie, Z., Han, Q., & Cai, Z. 2020, IEEE Internet of Things Journal, 8, 3088

Paruchuri, J., Cheung, S.-c., & Hail, M. 2009, EURASIP Journal on Information Security, 2009, 1

Pennington, J., Socher, R., & Manning, C. D. 2014, in Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 1532–1543

Phan, N., Wang, Y., Wu, X., & Dou, D. 2016, in AAAI, Vol. 16, AAAI, 1309–1316

Phan, N., Wu, X., Hu, H., & Dou, D. 2017, in 2017 IEEE International Conference on Data Mining (ICDM), IEEE, 385–394

Piersol, K. W., & Beddingfield, G. 2019, Pre-wakeword Speech Processing, uS Patent 10,192,546

Poria, S., Cambria, E., & Gelbukh, A. 2015, in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, ACL, 2539–2544

Poria, S., Cambria, E., Hazarika, D., Majumder, N., Zadeh, A., & Morency, L.-P. 2017a, in Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL, 873–883

Poria, S., Cambria, E., Hazarika, D., Mazumder, N., Zadeh, A., & Morency, L.-P. 2017b, in 2017 IEEE International Conference on Data Mining, IEEE, 1033–1038

Poria, S., Cambria, E., Howard, N., Huang, G.-B., & Hussain, A. 2016, Neurocomputing, 174, 50

Poria, S., Hazarika, D., Majumder, N., & Mihalcea, R. 2020, IEEE Transactions on Affective Computing

Rababah, A. 1993, Proceedings of the American mathematical society, 119, 803

Rahman, M. A., Rahman, T., Laganière, R., Mohammed, N., & Wang, Y. 2018, Trans. Data Priv., 11, 61

Rahmani, S., Hosseini, S., Zall, R., Kangavari, M. R., Kamran, S., & Hua, W. 2021, arXiv preprint arXiv:2106.14174

Rastogi, V., & Nath, S. 2010, in Proceedings of the 2010 ACM SIGMOD International

Conference on Management of data, ACM, 735–746

Raval, N., Machanavajjhala, A., & Cox, L. P. 2017, in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 1329–1332

Razavi, A., van den Oord, A., & Vinyals, O. 2019, CoRR, abs/1906.00446

Rosenberg, E. L., & Ekman, P. 2020, What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS) (Oxford University Press)

Roth, J. et al. 2019, arXiv preprint arXiv:1901.01342

Shokri, R., Stronati, M., Song, C., & Shmatikov, V. 2017, in 2017 IEEE Symposium on Security and Privacy (SP), IEEE, 3–18

Simonyan, K., & Zisserman, A. 2014, arXiv preprint arXiv:1409.1556

Soria-Comas, J., Domingo-Ferrer, J., Sánchez, D., & Megías, D. 2017, IEEE Transactions on Information Forensics and Security, 12, 1418

Sun, B., Feng, J., & Saenko, K. 2016in , AAAI

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. 2016, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2818–2826

Tang, W., Tan, S., Li, B., & Huang, J. 2017, IEEE Signal Processing Letters, 24, 1547

Truex, S., Liu, L., Gursoy, M. E., Yu, L., & Wei, W. 2018, arXiv preprint arXiv:1807.09173

Tsai, Y.-H. H., Bai, S., Liang, P. P., Kolter, J. Z., Morency, L.-P., & Salakhutdinov, R. 2019, in Proceedings of the Conference. Association for Computational Linguistics. Meeting, NIH Public Access, 6558

Tumasjan, A., Sprenger, T., Sandner, P., & Welpe, I. 2010, in Proceedings of the International AAAI Conference on Web and Social Media, Vol. 4, AAAI

Uittenbogaard, R., Sebastian, C., Vijverberg, J., Boom, B., Gavrila, D. M., et al. 2019, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 10581–10590

Ulutan, O., Rallapalli, S., Srivatsa, M., Torres, C., & Manjunath, B. 2020, in The IEEE Winter Conference on Applications of Computer Vision, 527–536

Verma, S. K. 2020, Method and system for sharing an output device between multimedia devices to transmit and receive data, uS Patent 10,581,933

Wang, C., Cai, Z., Seo, D., & Li, Y. 2023a, IEEE Internet of Things Journal

Wang, H., Xu, Z., Jia, S., Xia, Y., & Zhang, X. 2021, World Wide Web, 24, 1

Wang, J., Cai, Z., & Yu, J. 2019a, IEEE Transactions on Industrial Informatics, 16, 4242

Wang, J., Yang, Y., Mao, J., Huang, Z., Huang, C., & Xu, W. 2016a, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2285–2294

Wang, L., He, Z., & Cai, Z. 2023b, in 2023 IEEE 20th International Conference on Mobile Ad Hoc and Smart Systems (MASS), IEEE, 288–294

Wang, T.-C., Liu, M.-Y., Tao, A., Liu, G., Kautz, J., & Catanzaro, B. 2019b, arXiv preprint arXiv:1910.12713

Wang, T.-C., Liu, M.-Y., Zhu, J.-Y., Liu, G., Tao, A., Kautz, J., & Catanzaro, B. 2018, in Advances in Neural Information Processing Systems 31, 1144–1156

Wang, Y., Luo, B., Shen, J., & Pantic, M. 2019c, International Journal of Computer Vision, 127, 625

Wang, Y., Shen, Y., Liu, Z., Liang, P. P., Zadeh, A., & Morency, L.-P. 2019d, in Proceedings of the AAAI Conference on Artificial Intelligence, AAAI, 7216–7223

Wang, Y., Wu, X., & Hu, D. 2016b, in EDBT/ICDT Workshops, Springer, 0090–6778

Wang, Y., Wu, X., & Wu, L. 2013, in Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 329–340

Wang, Z., Chen, T., Li, G., Xu, R., & Lin, L. 2017, in Proceedings of the IEEE international conference on computer vision, 464–472

Wei, X.-S., Cui, Q., Yang, L., Wang, P., & Liu, L. 2019, arXiv preprint arXiv:1901.07249

Wu, H., Feng, J., Tian, X., Xu, F., Liu, Y., Wang, X., & Zhong, S. 2019a, in Proceedings of the 2019 Workshop on Hot Topics in Video Analytics and Intelligent Edges, ACM, 33–38

Wu, H., Zha, Z.-J., Wen, X., Chen, Z., Liu, D., & Chen, X. 2019b, in Proceedings of the 27th ACM International Conference on Multimedia, 620–628

Wu, N., Farokhi, F., Smith, D., & Kaafar, M. A. 2020, in 2020 IEEE Symposium on Security and Privacy (SP), IEEE, 304–317

Wu, X., Li, F., Kumar, A., Chaudhuri, K., Jha, S., & Naughton, J. 2017, in Proceedings of the 2017 ACM International Conference on Management of Data, 1307–1322

Xia, J., Huang, W., Ma, Z., Dai, X., & He, L. 2019, in 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT), IEEE, 208–215

Xiao, X., Wang, G., & Gehrke, J. 2010, IEEE Transactions on knowledge and data engineering, 23, 1200

Xiong, Z., Cai, Z., Han, Q., Alrawais, A., & Li, W. 2020, IEEE Transactions on Industrial Informatics, 14

Xiong, Z., Cai, Z., Hu, C., Takabi, D., & Li, W. 2022, IEEE Transactions on Dependable and Secure Computing

Xiong, Z., Cai, Z., Takabi, D., & Li, W. 2021a, IEEE Transactions on Industrial Informatics, 18, 1310

Xiong, Z., Li, W., & Cai, Z. 2023ain , 10537–10545

Xiong, Z., Li, W., Han, Q., & Cai, Z. 2019, in 2019 IEEE International Conference on Data Mining (ICDM), IEEE, 668–677

Xiong, Z., Li, W., Li, Y., & Cai, Z. 2023b, in The 23rd IEEE International Conference on Data Mining, IEEE

Xiong, Z., Xu, H., Li, W., & Cai, Z. 2021b, IEEE Transactions on Vehicular Technology, 70, 2822

Xu, C., Ren, J., Zhang, D., Zhang, Y., Qin, Z., & Ren, K. 2019, IEEE Transactions on Information Forensics and Security, 14, 2358

Xu, H., , Cai, Z., Xiong, Z., & Li, W. 2023a, in The 23rd IEEE International Conference on Data Mining, IEEE

Xu, H., Cai, Z., Li, R., & Li, W. 2022a, IEEE Transactions on Intelligent Transportation Systems, 23, 16600

Xu, H., Cai, Z., & Li, W. 2022b, ACM Transactions on Knowledge Discovery from Data (TKDD), 16, 1

Xu, H., Cai, Z., Takabi, D., & Li, W. 2021, IEEE Internet of Things Journal, 9, 1749

Xu, H., Li, W., & Cai, Z. 2023b, Theoretical Computer Science, 940, 90

Yin, X., Goudriaan, J., Lantinga, E. A., Vos, J., & Spiertz, H. J. 2003, Annals of botany, 91, 361

Yoon, J., Jordon, J., & van der Schaar, M. 2019, in International Conference on Learning Representations

Yu, W., Xu, H., Yuan, Z., & Wu, J. 2021in , 10790–10797

Zadeh, A., Chen, M., Poria, S., Cambria, E., & Morency, L. 2017, CoRR, abs/1707.07250

Zadeh, A., Liang, P. P., Mazumder, N., Poria, S., Cambria, E., & Morency, L.-P. 2018a, in Proceedings of the AAAI Conference on Artificial Intelligence, AAAI, 5634–5641

Zadeh, A., Liang, P. P., Poria, S., Vij, P., Cambria, E., & Morency, L.-P. 2018b, in Proceedings of the AAAI Conference on Artificial Intelligence, AAAI, 5642–5649

Zadeh, A., Zellers, R., Pincus, E., & Morency, L.-P. 2016, IEEE Intelligent Systems, 31, 82

Zadeh, A. B., Liang, P. P., Poria, S., Cambria, E., & Morency, L.-P. 2018c, in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL, 2236–2246

Zhang, G., Wang, C., Xu, B., & Grosse, R. 2018, arXiv preprint arXiv:1810.12281

Zhang, P., Thomas, T., & Emmanuel, S. 2012, Multimedia systems, 18, 175

Zhang, P., Thomas, T., Emmanuel, S., & Kankanhalli, M. S. 2010, in Proceedings of the 2nd ACM workshop on Multimedia in forensics, security and intelligence, 31–36

Zhang, S., Wen, L., Shi, H., Lei, Z., Lyu, S., & Li, S. Z. 2019a, International Journal of

Computer Vision, 127, 537

Zhang, T., Zhu, T., Xiong, P., Huo, H., Tari, Z., & Zhou, W. 2019b, IEEE Transactions on Industrial Informatics, 16, 2115

Zhang, Z., Shen, W., Qiao, S., Wang, Y., Wang, B., & Yuille, A. 2020, in The IEEE Winter Conference on Applications of Computer Vision, 1361–1370

Zheng, H., Hu, H., & Han, Z. 2020a, IEEE Intelligent Systems, 35, 5

Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., & Torr, P. H. 2015, in Proceedings of the IEEE international conference on computer vision, 1529–1537

Zheng, X., Cai, Z., & Li, Y. 2018a, IEEE Communications Magazine, 56, 55

Zheng, X., Luo, G., & Cai, Z. 2018b, IEEE Transactions on Network Science and Engineering, 7, 880

Zheng, X., Tian, L., Luo, G., & Cai, Z. 2020b, IEEE Internet of Things Journal, 7, 7883

Zhu, F., Li, H., Ouyang, W., Yu, N., & Wang, X. 2017, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5513–5522

Zhu, T., & Philip, S. Y. 2019, in 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), IEEE, 1601–1609