Novel Approaches for Multimedia Data Processing

A thesis presented in partial fulfilment of the requirements for

the degree of

Doctor of Philosophy

in

Computer Science

at Massey University, Albany, Auckland,

New Zealand.

Wanting Ji

2020

Abstract

Multimedia data processing is an active research field contributing to many frontiers of science and technology. It involves the processing of audio, image, video, text, and other forms of data. In this thesis, four novel approaches are proposed to address two key issues in multimedia data processing: (i) how to reduce the annotation costs of sound event classification/tagging, and (ii) how to improve the quality of video captions.

To address the issue of how to reduce the annotation costs of sound event classification/tagging, we propose a Gabor dictionary-based active learning (DBAL) approach for semi-automatic sound event classification. In DBAL, sound features are extracted from audio recordings through a Gabor dictionary. Based on the extracted features, sound events in the recordings will be manual or automatic tagged through active learning. Then a classifier is trained by these recordings with their true or predicted labels. Thus, DBAL can be evaluated by the accuracy of the classifier.

Further, a learnt dictionary-based active learning (LDAL) approach is proposed to tackle the same issue. In LDAL, a K-SVD learnt dictionary replaces the Gabor dictionary for feature extraction. The same active learning mechanism and classifier are used for tagging and evaluation. Compared with other existing approaches, our approaches (*i.e.*, DBAL and LDAL) achieve higher classification accuracies but require much fewer annotation costs.

To tackle the issue of how to improve the quality of video captions, we propose an attention-based dual learning (ADL) approach for video captioning. Two modules (*i.e.*, a caption generation module and a video reconstruction module) are contained in ADL, which are fine-tuned via dual learning. Thus, ADL can enhance the quality of the generated captions by minimizing the differences between raw and reconstructed/reproduced videos.

Further, we propose a bidirectional relational recurrent neural network (Bidirectional RRNN) to tackle the same issue. By fully utilizing the local and global context information as well as visual information in videos, Bidirectional RRNN can capture all events in a video, reason the relationships between events, and generate a set of

informative sentences to describe video contents. Experimental results on benchmark datasets demonstrate that our approaches (*i.e.*, ADL and Bidirectional RRNN) are superior to the state-of-the-art approaches.

In conclusion, this thesis proposes four effective approaches for processing multimedia data. Experimental results show that our approaches outperform the state-of-the-art approaches.

Acknowledgments

I would like to take this opportunity to express my deepest gratitude to all the people who supported me on my journey to achieve this qualification.

First of all, I would like to express my most sincere gratitude to my supervisor, Professor Ruili Wang, my co-supervisors, Professor Xun Wang and Dr Andrew Gilman, and other faculty members at the School of Natural and Computational Sciences, Massey University, who provided valuable academic guidance and spiritual support through my doctoral research. They spent dedicated time and effort in helping me to develop my research capabilities. It is enjoyable when discussing research problems with them. They provided not only constructive but also challenging feedbacks to improve my research work. Their intellectual knowledge and critical thinking deeply influenced my academic career. Without their valuable comments, suggestions, and persistent encouragement, it would be impossible for me to complete my doctoral research.

I am grateful to my friends and my colleagues in Professor Ruili Wang's research team for their sincere encouragement and valuable suggestions through my doctoral study.

I also would like to thank my parents for their unconditional love, understanding, and support.

I greatly acknowledge the funding from the China Scholarship Council towards my study and research.

Lastly, I would like to thank Massey University for its free academic atmosphere and broad academic platform, which broadened my horizon and extended my international academic experience.

Contents

List of Figures

List of Tables

Chapter 1 Introduction

This chapter provides an overview of this thesis. The motivations of this thesis are presented in Section 1.1. Then the scope of this thesis is presented in Section 1.2.

1.1 Motivations

Multimedia data processing is an active research field contributing to many frontiers of science and technology. It involves the processing of audio, image, video, text, and other forms of data. Recently, massive multimedia data, especially audio data and video data, is continuously being created and collected in different areas [1,2]. This attracts more and more researchers to develop various approaches to deal with the growing multimedia data. These sophisticated and robust approaches will provide great unprecedented opportunities to overcome the challenges and issues in multimedia data processing [1]. Therefore, digital audio processing and digital video processing have become two important research subfields of multimedia data processing. There is a great need to develop novel approaches to deal with different tasks in these two research fields.

In this chapter, a brief introduction to digital audio processing and digital video processing is presented in Sections 1.1.1 and 1.1.2, respectively.

1.1.1 Digital audio processing

Digital audio processing focuses on processing digital audio recordings using various computing methodologies [3]. Specifically, in the real world, digital audio recordings can be widely derived and collected from speech, music, environmental sounds, and even artificial synthetic data. According to specific digital audio processing tasks, the collected digital audio recordings will be processed/manipulated in a variety of ways, including editing (*e.g.*, trim, split, and merge), enhancing (*e.g.*, amplify and denoise), analyzing (*e.g.*, visualize and classify), and creating effects (*e.g.*, pitch shift and add reverb) [4].

Since a digital audio recording is usually composed of data such as speech, music, and sounds, digital audio processing can be roughly divided into three categories: speech processing, music processing, and sound processing.

- Speech processing. Speech recognition and speech synthesis are two main tasks in speech processing [5]. Speech recognition aims to develop methodologies that can recognize and translate spoken language into text, while speech synthesis is to convert natural language text into speech. Recent research on these two tasks has made remarkable progress. Various speech recognizers and synthesizers are widely used in many real-world applications, such as banking services and telephone services.

- Music processing. Music processing aims to analyze, manipulate, and synthesize music data for specific tasks [6]. Recent research on music processing involves extracting meaningful features from music data, and then integrating/combining these features with other information sources (*e.g.*, lyrics, sheet music, and contextual metadata obtained by collaborative tagging or expert annotators) for different music processing tasks, such as music information retrieval and music recommendation.

- Sound processing. Sound event classification and sound event detection are two main tasks in sound processing. Sound event classification aims to recognize a set of active sound events in audio recordings. In addition to classification, sound event detection also requires detecting the temporal onset and offset of each sound event in the audio recordings [7]. Recent research on these two tasks has been widely applied to real-world applications, such as surveillance systems.

In recent years, with the development of machine learning techniques, especially the development of deep learning techniques, in-depth research has been conducted on speech processing tasks and music processing tasks. For example, the most advanced speech recognition model is a deep learning model based on self-attention [8], which is trained in an end-to-end manner. It removes all intermediate steps and independent subtasks of traditional speech recognition models (*e.g.*, hidden Markov models), and can achieve high recognition accuracy. The word error rates (WER) on the test set can be reduced to about 10% [8].

Compared with speech processing, music processing is a relatively young but rapidly growing research field. Recently, a variety of research related to music processing, including music information retrieval [9], music computing [10], audio-effects processing [11], and applications in audio engineering [12], have achieved remarkable results.

Contrary to music processing, although sound processing is a research field with a

long tradition, there are still many challenges that need to be addressed. For example, since the sound event classification task can be treated as a supervised classification task, sound event classifiers usually require a large number of sound segments with their true labels for training [7]. However, in the real world, since the annotation (*i.e.*, manual tagging sound events) cost is much expensive and time-consuming, the number of labeled sound segments is limited. Thus, there is a great need to develop novel sound event classifiers that can achieve comparable or higher classification accuracy but requires fewer labeled sound segments for classifier training. Therefore, the first issue to be addressed in this thesis is *how to reduce the annotation cost in environmental sound event classification/tagging*.

To address this issue, this thesis proposes two semi-automatic sound event classification/tagging approaches, which achieve comparable or higher classification accuracy but require fewer annotation costs. The details of these two approaches are presented in Chapters 2 and 3, respectively.

1.1.2 Digital video processing

Digital video processing is another important research subfield of multimedia data processing. Since video data can be considered as a series of time-varying images, digital video processing tasks encompasses many tasks derived from the essential principles of digital image processing (*e.g.*, computer graphics tasks), as well as some tasks that exploit the temporal nature of video data (*e.g.*, computer vision tasks) [13].

As shown in Figure 1.1, based on a variety of computing methodologies, digital video processing tasks can be divided into two categories: computer graphics tasks, and computer vision tasks. Most computer graphics tasks focus on still image information in videos and can be processed using digital image processing approaches, such as image compression approaches and image enhancement approaches. Computer vision tasks include video segmentation task, video tracking task, video captioning task, and many others. In these tasks, image information, temporal or motion information, and even audio information in videos will be captured and processed, which cannot be handled by digital image processing approaches.

Among all computer vision tasks, video captioning is an emerging task that aims to generate sentences/captions to describe video content [14-19]. Real-world

applications based on video captioning approaches (*e.g.*, assistant services and robots) have been widely used in human life, and provide many conveniences for human life. For example, visually impaired people can know what is happening around them through video captioning based glasses. However, the video captions generated by the existing video captioning approaches are still not comparable to human-generated descriptions. Thus, there is a great need to develop novel video captioning approaches that can generate high-quality video captions to describe video content. Therefore, the second issue to be addressed in this thesis is *how to improve the quality of video captions*.

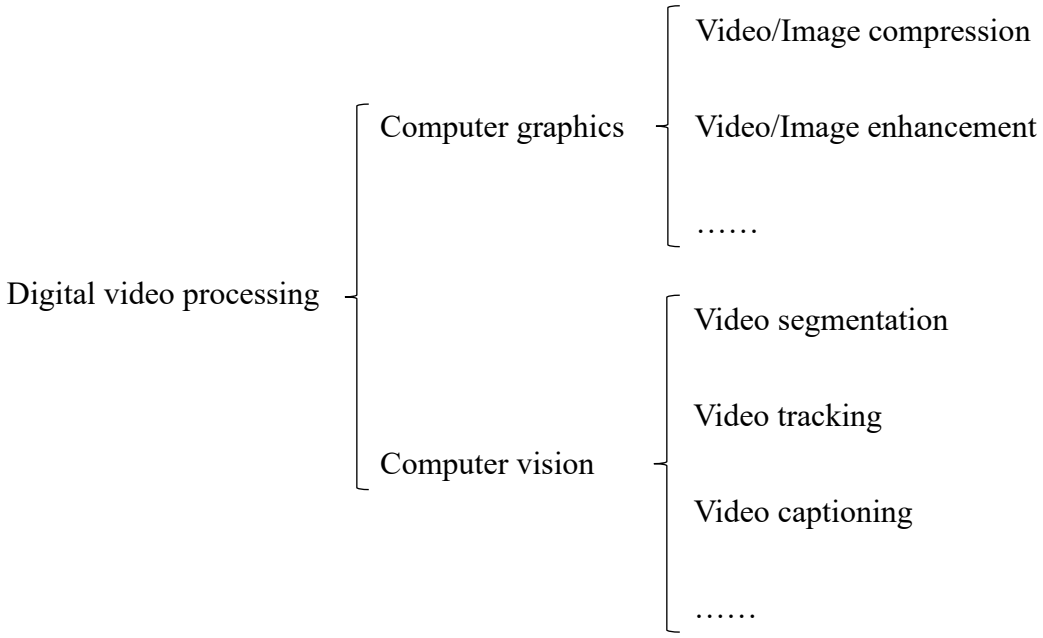


Figure 1.1. Categories of digital video processing tasks.

To address this issue, this thesis proposes a video captioning approach in Chapter 4, which can generate a single-sentence caption to describe the main content of a video. Then, to further generate captions for the video containing multiple video events, a dense video captioning approach is proposed in Chapter 5, which can detect/capture all video events and generate dense video captions (*i.e.*, a set of sentences).

1.1.3 Summary

In summary, multimedia data processing is a booming research field, which covers a vast of subfields and applications such as digital audio processing and digital video processing. This thesis tackles two key issues in multimedia data processing: (i) how

to reduce the annotation costs of sound event classification/tagging, and (ii) how to improve the quality of video captions. To address these two issues, four novel approaches are proposed.

1.2 Scope of this thesis

This thesis is organized as follows:

Chapter 2: developing a novel *Gabor dictionary-based active learning* (*DBAL*) *approach* for environmental sound event classification. In DBAL, a Gabor dictionary is used to extract features from audio recordings. Based on the extracted features, the way of tagging sound events in the recordings (*i.e*., manual or automatic tagging) is selected through active learning. Then a sound event classifier is trained by these recordings with their manual or predicted labels. Thus, the performance of DBAL can be evaluated by the accuracy of the classifier. Compared with other sound event classification approaches, DBAL achieves comparable classification accuracy but requires fewer annotation costs.

Chapter 3: developing a novel *learnt dictionary-based active learning* (*LDAL*) *approach* for environmental sound event tagging. In LDAL, a K-SVD learnt dictionary replaces the Gabor dictionary for feature extraction. The same active learning mechanism and classifier used by the DBAL approach are utilized to assign labels to the sound events in audio recordings and to evaluate the classification accuracy of the proposed LDAL approaches. Compared with the DBAL approach, LDAL achieves higher tagging accuracy in the case of the same annotation costs.

Chapter 4: developing a novel *attention-based dual learning* (*ADL*) *approach* for video captioning. Two modules (*i.e*., a caption generation module and a video reconstruction module) are contained in ADL to leverage the information in raw videos and the generated captions. Then a multi-head attention mechanism is used to help the two modules attend to the most effective information in videos and captions, and a dual learning mechanism is used to fine-tune the performance of the two modules. Thus, ADL can improve the quality of the generated captions by minimizing the differences between raw and reconstructed/reproduced videos.

Chapter 5: developing a novel *bidirectional relational recurrent neural network* (*Bidirectional RRNN*) for dense video captioning. In Bidirectional RRNN, a bidirectional RRNN encoder, which has a relational memory core for collecting and

relational reasoning of temporal context information, is proposed to obtain the local and global context information of a target event. Thus, the proposed approach can capture all events in videos, reason the relationships between these events, and then generate a set of informative sentences to describe video contents.

In summary, this thesis proposes four novel approaches to correspondingly address the two key issues in multimedia data processing, which are tackled in Section 1.1, *i.e.*, (i) how to reduce the annotation costs of sound event classification/tagging, and (ii) how to improve the quality of video captions. The approaches proposed in Chapters 2 and 3 have already been published in journal papers [20,21], respectively. The approaches proposed in Chapters 4 and 5 have been submitted to a conference [22] and a journal [23] in the form of papers, respectively. Note that references related to each chapter are listed at the end of each chapter.

References

1. J. Song, H. Jegou, C. Snoek, Q. Tian, and N. Sebe. Guest editorial: Large-scale multimedia data retrieval, classification, and understanding. *IEEE Transactions on Multimedia*, 19(9): 1965-1967. 2017.
2. L. Gao, J. Song, X. Liu, J. Shao, J. Liu, and J. Shao. Learning in high-dimensional multimedia data: the state of the art. *Multimedia Systems*, 23(3): 303-313. 2017.
3. S. K. Gaikwad, B. W. Gawali, and P. Yannawar. A review on speech recognition technique. *International Journal of Computer Applications*, 10(3): 16-24. 2010.
4. Wolfram language & system documentation centre: Audio processing. Available at: https://reference.wolfram.com/language/guide/AudioProcessing.html
5. S. Furui. Digital speech processing: synthesis, and recognition. *CRC Press*, 2018.
6. M. Mueller, B. A. Pardo, G. J. Mysore, and V. Valimaki. Recent advances in music signal processing. *IEEE Signal Processing Magazine*, 36(1): 17-19. 2018.
7. H. Fayek, V. Tourbabin, and S. Adavanne. Sound event classification and detection with weakly labeled data. In *Detection and Classification of Acoustic Scenes and Events*, pp. 15-19. 2019.
8. N. Pham, T. Nguyen, J. Niehues, M. Müller, and A. Waibel. Very deep self-attention networks for end-to-end speech recognition. In *Interspeech*, pp. 66-70. 2019.
9. M. Müller, A. Arzt, S. Balke, M. Dorfer, and G. Widmer. Cross-modal music retrieval and applications: An overview of key methodologies. *IEEE Signal Processing Magazine*, 36(1): 52-62. 2018.
10. E. Benetos, S. Dixon, Z. Duan, and S. Ewert. Automatic music transcription: An

overview. *IEEE Signal Processing Magazine*, 36(1): 20-30. 2018.

11. D. A. D'Souza, and V. V. D. Shastrimath. Modelling of audio effects for vocal and music synthesis in real time. In *the 3rd International Conference on Computing Methodologies and Communication*, pp. 1-4. 2019.
12. J. Barbour. Spatial audio engineering: exploring height in acoustic space. *RMIT University, Australia*. 2017.
13. A. M. Tekalp. Digital video processing. *Prentice Hall Press*, 2015.
14. J. Wang, W. Wang, Y. Huang, L. Wang, and T. Tan. M3: Multimodal memory modeling for video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7512-7520. 2018.
15. B. Wang, L. Ma, W. Zhang, and W. Liu. Reconstruction network for video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7622-7631. 2018.
16. J. Xu, T. Yao, Y. Zhang, and T. Mei. Learning multimodal attention LSTM networks for video captioning. In *Proceedings of the 25th ACM International Conference on Multimedia*, pp. 537-545. 2017.
17. R. Xu, C. Xiong, W. Chen, and J. J. Corso. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *the 29th AAAI Conference on Artificial Intelligence*, pp. 1-10. 2015.
18. T. Baltrušaitis, C. Ahuja, and L. P. Morency. Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2): 423-443. 2018.
19. V. Ramanishka, A. Das, D. H. Park, S. Venugopalan, L. A. Hendricks, M. Rohrbach, and K. Saenko. Multimodal video description. In *Proceedings of the 24th ACM International Conference on Multimedia*, pp. 1092-1096. 2016.
20. W. Ji, R. Wang, and J. Ma. Dictionary-based active learning for sound event classification. *Multimedia Tools and Applications*, 78(3): 3831-3842. 2019.
21. X. Qin, W. Ji (corresponding author), R. Wang, and C. Yuan. Learnt dictionary based active learning method for environmental sound event tagging. *Multimedia Tools and Applications*, 78(20): 29493-29508. 2019.
22. W. Ji, R. Wang, Y. Tian, and X. Wang. An attention based dual learning approach for video captioning. *Neurocomputing*, submitted. 2020.
23. W. Ji, R. Wang, and M. Liu. Dense video captioning with context fusion and reasoning. *Image and Vision Computing*, submitted. 2020.

Chapter 2 A Gabor dictionary-based active learning approach

This chapter proposes a new Gabor dictionary-based active learning (DBAL) approach for sound event classification, which significantly reduces the required number of labeled samples in the process of sound event classifier training. In DBAL, since dictionary learning is more suitable for sound feature extraction/representation, a Gabor dictionary is used to extract features from audio recordings. Then an active learning mechanism, which is achieved through clustering, is used to select the way of sound event labeling (*i.e.*, assign true labels or predicted labels to sound segments). Our sound event classifier will be trained using sound segments with their true and/or predicted labels. We test DBAL and other reference approaches on a public urban sound dataset with 8732 sound segments. The classification accuracy is used to measure the performance of these approaches. Experimental results show that our approach has higher classification accuracy but requires much fewer annotation costs than other approaches.

This chapter is organized as follows. Section 2.1 introduces the background of sound event classification and the motivation of this research, and then reviews the most relevant works of this research. Section 2.2 presents the proposed sound event classification approach in detail. Section 2.3 presents the experiments and discusses the experimental results. At the end of this chapter, the conclusion of this chapter is presented in Section 2.4.

2.1 Introduction

Sound event classification is a process that involves classifying input audio signals based on their salient features/characteristics, which plays an essential role in identifying, analyzing, and utilizing the environmental sound information under a background sound. Over the past few years, sound event classification [19,24,25,30] has gained much interest in the field of audio signal processing [9,12,15,28,31,34,36] and has been widely applied to noise detection [1], monitoring [6,13] and other real-world applications.

In practice, there are a limited number of labeled training samples whereas unlabeled training samples are easily available. The shortage of labeled samples is one of the

main challenges and hindrances in the training process that affects sound event classifiers, which in turn limits the classification accuracy in real situations [4,5,17]. Through literature review, it was easy to find that even the largest environmental sound dataset ESC-US [18] only contained a limited number of labeled sound samples (2000 recordings) and a large number of unlabeled sound samples (250,000 recordings). This situation can be attributed to the expensive labeling process (*i.e.*, assigning a predefined label to a sound sample), which is particularly pronounced in large datasets [18]. Therefore, it is necessary to develop novel techniques that make full use of both labeled samples and unlabeled samples in the process of sound event classifier training.

Semi-supervised learning is one of the effective approaches for such scenarios [35], which uses a small number of labeled samples and a large number of unlabeled samples in the classifier training process. Zhang *et al*. proposed a semi-supervised framework for sound event classification [32]. It first took the confidence of a classifier in five levels for classification. Then they added re-sampled originally labeled samples and unlabeled samples, which had a high confidence level to the training datasets. An iterative process was used to enhance classification accuracy. However, the labeled samples were preselected in semi-supervised learning. In some cases, these labeled samples cannot reflect the true situation of the whole dataset.

Active learning is a special case of semi-supervised learning. It aims at achieving higher accuracy with fewer training labels by (actively) choosing samples from which it learns when the annotation cost is expensive. Since active learning can select the most informative and representative samples to be labeled by the learner, it reduces annotation costs [20]. Thus, active learning can achieve the maximum gain in learning by using a small number of labeling queries [7].

An active learning approach called Medoid-based Active Learning (MAL) was proposed for urban sound event classification in [25]. The MAL approach extracted features from sound samples using Mel-frequency Cepstral Coefficients (MFCCs). Active learning in MAL was used to label the input sound samples. However, since MFCCs perform better on structured sounds such as speech and music rather than on noise-like sound recognition such as environmental sounds [2], better feature extraction methods should be used or developed for sound event classification.

Furthermore, a combination of active learning and self-training sound event classification approach was proposed in [10]. Initially, all sound samples were

unlabeled. Then they calculated the classifier confidence scores of these unlabeled sound samples. The classifier confidence score is a probability to measures the classifier's output certainty level, which indicates the classifier's confidence about the predicted labels. The samples which had lower confidence scores were labeled by the learner, while the high-score samples were labeled by using a self-training approach automatically. Finally, all sound samples with their labels were used to train a sound event classifier.

In recent years, multiple dictionary learning methods have been developed such as Wavelets [27], Cosine packets [26], Gabor dictionaries [14,23], and Chirplets [8]. Based on adaptive approximation techniques such as Matching Pursuit (MP) [2] and Orthogonal Matching Pursuits (OMP) [2], a new representation of the sound input samples in the form of the linear combination of basic atoms from a dictionary can be obtained [2,21]. Several dictionary learning methods such as Fourier [8], Haar [7], and Gabor have been evaluated with the MP method for environmental sound event classification in [2]. Based on their experimental results, the Gabor dictionary achieves better classification results than other dictionaries. Wang *et al*. also proposed a Gabor-based environmental sound event classification in [29]. In their approach, input samples were firstly represented using the atoms in a Gabor dictionary. Then, Principal Component Analysis (PCA) and Linear Discriminate Analysis (LDA) were used to set up a scale-frequency map to generate features. The represented features were used to train a classifier for sound event classification.

In this chapter, a new Dictionary-Based Active Learning (DBAL) approach is proposed for sound event classification. The proposed approach combines dictionary learning [29] for feature extraction and active learning [25] for labeling, which significantly reduces the required number of labeled samples for the processing of sound-classifier training. The proposed approach consists of three phases, named feature extraction, active learning-based labeling, and classification, respectively. In the process of feature extraction, all training samples (*i.e*., sound segments) are represented using the atoms in a Gabor dictionary. After that, the *k*-medoids clustering method is used to cluster the represented training samples. The medoids will be labeled with their true labels, while other members in the same clusters will be labeled with predicted labels. This is an active learning process. Finally, the represented training samples with their (either true or predicted) labels are used to train a sound event classifier. The classification accuracy of this classifier is used to measure the performance of the proposed approach. Figure 2.1 illustrates the procedure of the proposed approach.

The proposed DBAL approach is compared with other approaches on a public urban sound dataset with 8732 sound segments in 10 sound categories. The experimental results show that when the proposed approach has the same or better accuracy than other reference approaches, it requires a much less annotation cost.

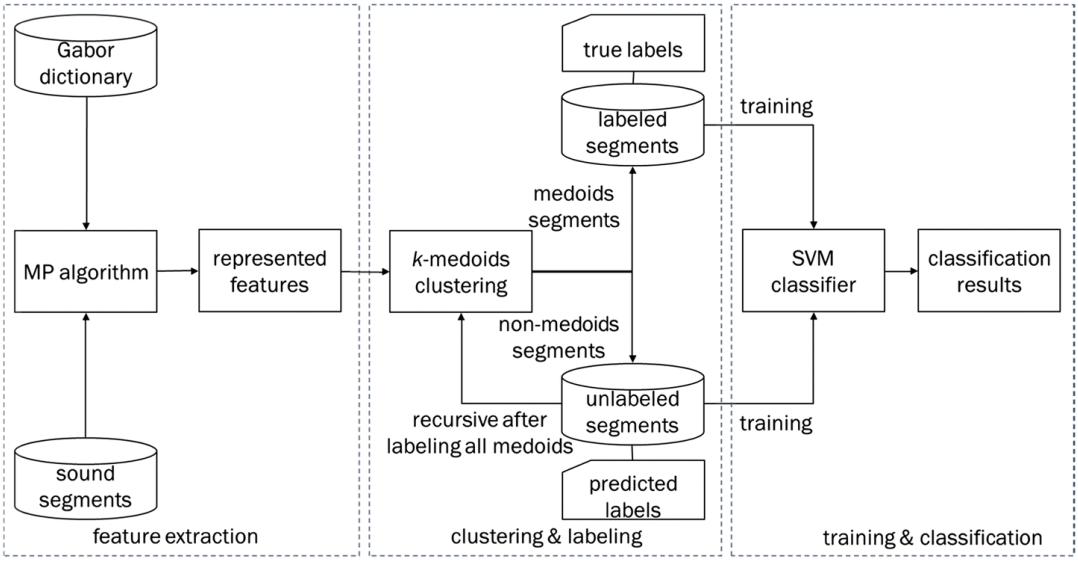


Figure 2.1. Flowchart of the proposed DBAL semi-automatic sound event classification approach.

2.2 The proposed approach

As shown in Figure 2.1, three phases (i.e., feature extraction phase, clustering-based actively labeling phase, and sound event classification phase) are contained in the proposed approach. In the following sections, we will present our proposed sound event classification approach in detail.

2.2.1 Feature extraction

In recent years, many different feature extraction methods (*e.g.*, MFCCs, band energy ratio, zero crossing rates, and dictionary representation) have been developed to represent environmental sound events. Due to the following reasons, this chapter constructs a Gabor dictionary to represent the whole sound samples: (i) Gabor dictionary performs better on sound event classification by extracting time-frequency domain features while other feature extraction methods such as MFCCs only extract features in the frequency domain [2]. (ii) MFCCs perform better on

structured sounds such as speech and music instead of on noise-like sound recognition such as environmental sounds [2].

In this chapter, the proposed approach firstly selects several atoms in a Gabor dictionary to approximate an input sound sample using the MP algorithm. Each atom in the dictionary is in the form of the given Gabor function, which consists of a scale, position, frequency, and phase information.

Gabor dictionary selection. Gabor functions are sine-modulated Gaussian functions that are scaled and translated to provide joint time-frequency positions. The mathematical definition of a discrete Gabor time-frequency atom is expressed as:

$$G_{s,u,f,\theta}(t) = \frac{K_{s,u,f,\theta}}{\sqrt{s}} e^{-\pi(t-u)^2/s^2} cos[2\pi f(t-u)+\theta], \quad (2.1)$$

where s represents the scale corresponded to the width of the Gabor function in time; u denotes the central temporal position of the Gabor function; f refers to the frequency; θ refers to the phase; t denotes the indices of the sampling points of an input sound segment; $K_{s,u,f,\theta}$ is a normalization factor such that $\| G_{s,u,f,\theta} \|^2 = 1$.

According to the experimental results in [2] and [29], the following parameters are selected: $s = \{2^p \mid p = 1, 2, \ldots, 8\}$; $u = \{0, 64, 128, 192\}$; $f = \{150, 450, 840, 1370, 2150, 3400, 5800\}$Hz; $\theta = 0$; the atom length is truncated to $T = 256$ so that $t = 1, 2, \ldots, T$. Thus, a Gabor dictionary is constructed based on these parameters, *i.e.*, 224 atoms (8 levels of scale × 4 central positions × 7 frequencies) are in the Gabor dictionary.

Matching pursuit algorithm. When the Gabor dictionary was established, a training sample can be represented with Gabor atoms by using the MP algorithm. The first MP algorithm is proposed in [14]. It decomposes the input sound signal by using the atoms in an overcomplete dictionary and provided a sparse linear expansion of the input signal.

An MP algorithm consists of two phases: the selection and decomposition phases. In the selection phase, the MP algorithm selects every atom from the current dictionary to check the close similarity between this atom and the input sound signal by computing the inner products between them.

Assume that dictionary D is a collection of atoms given by:

$$D = \{\phi_\gamma : \gamma \in \Gamma\}, \tag{2.4}$$

where Γ denotes the parameter set and ϕ_γ denotes an atom. Then the approximate decomposition of the input signal s can be represented as:

$$s = \sum_{i=1}^{n} \alpha_{\gamma_i} \phi_{\gamma_i} + R^{(n)}, \tag{2.5}$$

where $R^{(n)}$ denotes the residual signal; $\alpha_{\gamma i}$ denotes the coefficient of $\phi_{\gamma i}$; n denotes the number of atoms used to represent s. Initially, $R^{(0)} = s$, and the MP algorithm calculates all inner products of s with the atoms in D. We select the atom with the largest magnitude inner product $\phi_{\gamma 0}$ as the first element. Mathematically, this can be represented as:

$$\left|\langle s, \phi_{\gamma_0} \rangle\right| \geq \left|\langle s, \phi_\gamma \rangle\right|,$$

where <·> denotes the inner product operation and |·| denotes the absolute operation. Then the atom $\phi_{\gamma 0}$ is subtracted from s to get residual $R^{(0)}$. In this way, the approximation of s at the i^{th} step can be calculated by:

$$s^{(i)} = s^{(i-1)} + \alpha_i \phi_{\gamma_i}, \tag{2.6}$$

where $\alpha_i = \langle R^{(i-1)}, \phi_{\gamma_i} \rangle$ and $R^{(i)} = s - s^{(i)}$. After n steps, the stop criterion of the MP algorithm is reached. Since the best atom (*i.e.*, the atom has the largest absolute inner products with the input signal) is chosen every time in the selection phase, the reconstruction error is minimal when the selected atom used to represent the input signal.

2.2.2 Clustering and labeling

In the processing of labeling, the k-medoids clustering is used on the training samples with their Gabor atom representations. The medoids of clusters get their true labels while others are assigned with their predicted labels by using the proposed approach below. Repeat this process if the annotation cost is larger than the number of cluster k.

Dissimilarity distance. There are two dissimilarity measurements used in clustering analysis. One is feature projection, which reflects similarity relation between two objects; the other is distance calculation, which reflects the difference between two objects such as Kullback-Leibler (KL) divergence. In general, the KL divergence between two discrete probability distributions is defined as follows:

$$D_{KL}(f||g) = \sum_{x \in D} f(x) log(f(x)/g(x)), \tag{2.7}$$

where f and g are two probability functions in a discrete domain D with a finite number of values. In this chapter, the KL divergence is used to measure dissimilarity between two sound samples. Because the KL divergence is an asymmetric operation so that $D_{KL}(f \| g)$ is different from $D_{KL}(g \| f)$. However, the dissimilarity between two sound samples is always the same. Thus, the dissimilarity between two sound samples in this chapter is defined as:

$$\widetilde{D}_{KL}(f||g) = \widetilde{D}_{KL}(g||f) = \frac{D_{KL}(f||g)+D_{KL}(g||f)}{2}. \tag{2.8}$$

Algorithm 2.1 *k*-medoids clustering algorithm

Input: number of clusters *k*, training object set $\alpha = \{\alpha_i\}_{i=1,\dots,m} \in \mathbb{R}^t$

Output: *k* cluster set $C = \{C_1, \dots, C_k\}$

(1) randomly select *k* objects as initial medoids $O = \{o_1, \dots, o_k\}$;

(2) calculate the Euclidian distance between every pair of objects;

(3) assign other objects to the nearest medoid in *O*;

(4) **for** each medoid o_* and the objects associated with o_*:

accumulate the dissimilarity distances between o_* and the objects associated with o_*;

find a new medoid $\widetilde{o_*}$, the accumulated distance between the new medoid $\widetilde{o_*}$ and other objects in the cluster are minimal;

update the current medoid o_* by replacing with the new medoid $\widetilde{o_*}$;

(5) repeat step (3) and (4) alternately until there is no change in *O*;

(6) **return** *k* cluster set $C = \{C_1, \dots, C_k\}$;

(7) **end**

***K*-medoids clustering**. Clustering is a task of grouping a set of data objects into clusters so that the data objects in the same cluster are more similar to each other but much different from the data objects in the other clusters [16]. The *k*-medoids clustering is a centroid-based clustering process, which finds the k medoids

iteratively and assigns other data objects to the nearest medoid, where a medoid is a data object in the dataset. Because it is based on the most centrally placed data objects in a cluster, it is less sensitive to outliers than the *k*-means clustering [16]. The process of the *k*-medoids clustering algorithm is shown in Algorithm 2.1.

The proposed approach uses the furthest-first traversal method to select initial medoids, which can effectively avoid local redundancy problems. A traversed set starts with a randomly selected sound segment. The distances between every pair of all sound segments are calculated by the chosen dissimilarity measurement (*i.e.*, the KL divergence in this case). Then, it is updated iteratively to minimize the total distance of all sound segments to the nearest medoids until no medoid can be swapped to reduce the total distance. The sound segment located at the centroid of each cluster is the medoid of each cluster.

DBAL aims to achieve the same or higher accuracy than the other approaches while use less labeled training samples. Thus, different from general *k*-medoids clustering, the proposed approach attempts to reduce the average size of clusters using a larger number of clusters so that the training samples in a cluster are more similar to each other. In this chapter, in order to compare with reference approaches, the number of clusters *k* in the proposed approach is set to $k = m/4$ (the same setting is used in MAL [25]), where *m* is the number of unlabeled training samples.

Algorithm 2.2 DBAL algorithm

Input: *m* sound segment samples, Gabor dictionary *D*, the annotation cost *sum*

Output: sample label set $L = \{l_1, \ldots, l_m\}$

(1) extract features from m sound segment samples;

(2) represent features using the atoms in Gabor dictionary D;

(3) determinate the initial number of clusters $k=m/4$;

(4) **do** *k*-medoids clustering on the represented samples;

(5) assign true or predicted labels to samples according to the relationship between *sum* and *k*;

(6) **return** *L*;

(7) **end**

Label assignment and recursive process. The processing of labeling can be described as followed. Initially, all sound segments are unlabeled. Then the *k*-medoids clustering is used to cluster these samples based on their new representations. After the *k*-medoids clustering, the medoid of each cluster will be

labeled with its true label, while other members in the same cluster will get the same labels as their medoid's label as their predicted label. Since the annotation cost (*i.e.*, the number of true labels) is predefined, thus, (i) if there are more than k true labels can be labeled (*i.e.*, the annotation cost is larger than k), the process of the k-medoids clustering will be made recursively; (ii) if there are less than k true labels can be labeled (*i.e.*, the annotation cost is less than k), all clusters will be sorted in descending order of size so that the clusters with more members will be labeled first, which can allow more unlabeled samples will get their predicted labels. The whole process is shown in Algorithm 2.2.

The goal of the proposed approach is to label all training samples with the least annotation cost rather than assign more true labels by using a larger annotation cost to get a higher labeling accuracy. Thus, the largest cluster is labeled first in each k-medoids clustering, which means a large number of predicted labels will be assigned at each time to reduce the annotation cost.

2.3 Experiments

The training samples with a true label or predicted labels are used for training a supervised multi-class SVM. The classification accuracy of DBAL is used to evaluate the performance of the proposed approach.

2.3.1 Dataset

To validate the proposed approach, all experiments are tested on the UrbanSound8K dataset [22], which is a public urban sound dataset that includes 8732 labeled sound segments (<=4s) of urban sounds from 10 classes: air_conditioner, car_horn, children_playing, dog_bark, drilling, enginge_idling, gun_shot, jackhammer, siren, and street_music. The whole dataset is pre-sorted into ten folds for cross-validates the automatic classification results (Table 2.1).

2.3.2 Experiment setup

In the experiments, each sound segment in the UrbanSound8K dataset makes up an instance for training or testing. The proposed approach uses a frame window of 256 points with a 50% overlap to divide training samples. Using the MP algorithm, every training sample will be represented using Gabor atoms. The following summary statistics of Gabor atoms are used as segment-wise features: maximum, minimum,

medium, mean, standard deviation, skewness, and kurtosis.

Table 2.1. Sound event classification in the UrbanSound8K dataset.

Taxonomy	Number of sound segments	Total duration(s)
air_conditioner	1000	3994.9287
car_horn	429	1053.9532
children_playing	1000	3961.8745
dog_bark	1000	3148.7495
Drilling	1000	3548.2440
enginge_idling	1000	3935.9925
gun_shot	374	616.7964
Jackhammer	1000	3610.9747
Siren	929	3632.7015
street_music	1000	4000
Total	8732	31,504.2155

In each round of evaluation, 90% of sound segments in the dataset are used for training, while 10% of sound segments are used for testing. The labels provided by the dataset are used as ground truth. In a training set, all true labels are hidden initially. In the processing of using the proposed approach, the annotation cost sum can assign up to sum true labels for labeling.

A supervised segment-level multiclass SVM is used as a classification model. The classifier is trained by the input sound segments (*i.e.*, training samples) with their produced (*i.e.*, either true or predicted) labels. Since the proposed approach does not aim to the optimize classification model or the SVM method, the parameters of this classification model come from the default setting of SVM in Python Scikit-learn (http://scikit-learn.org/stable/index.html). Other classification methods such as random forest classifiers and decision tree classifiers are also used in the experiments with the default setting in Python Scikit-learn. The classification accuracies of standard random forest classifiers and standard decision tree classifiers are similar to the results of SVM [22,37-40]. Thus, the SVM results are used here. All the experiments are repeated five times and the averaged results are reported.

2.3.3 Reference approaches

Semi-Supervised Learning (SSL) for sound event classification is used as the first reference approach, which selects samples for semi-supervised learning by using

random sampling [25]. The number of selected samples is the annotation cost. The random sampling is used for simulating the performance of passive learning [11].

The second reference approach is certainty-based active learning (CRTAL) [20], which has been used for speech recognition [3]. In the processing of active learning, half of the annotation cost is used for sample selection, which selects samples randomly. The other half of the annotation cost is used for uncertainty selection. The batch size is five. In each iteration, the least confident five samples to the current system are assigned with labels. Then the system is updated by adding new labels to the training sets.

MAL [25] is used as the last reference approach. Similar to the proposed approach, it uses active learning for labeling input samples and trains a sound event classifier according to these input samples with their obtained labels. The features used in the active learning are summary statistics of MFCCs, which include minimum, maximum, median, mean, variance, skewness, kurtosis, and the median and variance of the first and second derivatives.

2.3.4 Experimental results

Figure 2.2 shows the performance of DBAL compared with other reference approaches. With the annotation cost increase, the classification accuracy of the sound event classifier raises nonlinearly. All sound segments in the training dataset will get their true labels or predicted labels. The accuracy of the obtained sound event classifier can achieve 67% when we assume that all predicted labels assigned to the sound segments are consistent with their true labels.

According to Figure 2.2, when the annotation cost is the same, the obtained classifier in the DBAL approach achieves higher classification accuracy than any other reference approaches. Compared with DBAL, other reference approaches need 1.5–4 times of annotation cost to achieve the same accuracy. In other words, the proposed approach outperformed other reference approaches to achieve the same accuracy by using the least annotation cost.

When DBAL assigns 1700 true labels for labeling, every sound segment in the training dataset can get either a true label or predicted labels. However, other reference approaches need more than 2000 true labels for labeling to achieve approximate classification accuracy.

Figure 2.2. The classification accuracy of DBAL and other reference approaches.

2.4 Summary

This chapter develops a new Gabor dictionary based active learning (DBAL) approach for sound event classification. Initially, the proposed approach selects 5 atoms from a pre-constructed Gabor dictionary to approximate input sound segments (*i.e.*, training samples) using the MP algorithm. Then these training samples will be clustered using the *k*-medoids clustering approach repeatedly. During the clustering, the medoids of clusters will be labeled with their true labels while other sound segments in the clusters will be labeled with their predicted labels. Finally, these input sound segments with their true labels or predicted labels are used to train a multi-class sound event classifier.

The proposed DBAL approach is tested on the UrbanSound8k dataset, which includes 10 categories of real-life urban sound segments. The accuracy of the sound event classifier is used to evaluate the performance of the DBAL approach. According to the experimental results, DBAL achieved a classification rate of 67% in the unknown environments, without any preprocessing or prior knowledge of the noise, while the annotation cost is 1700. The extensive experimental comparisons show that the DBAL approach outperformed other reference approaches in terms of classification accuracy but uses less annotation cost in the task of sound event classification. In the future, we will attempt to use other classifiers like [33] or deep learning in the processing of classification to improve the accuracy of the classifier.

References

1. B. D. Barkana, and B. Uzkent. Environmental noise classifier using a new set of feature parameters based on pitch range. *Applied Acoustics*, 72(11): 841-848. 2011.
2. S. Chu, S. Narayanan, and C. J. Kuo. Environmental sound recognition with time-frequency audio features. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(6): 1142-1158. 2009.
3. D. Cohn, L. Atlas, and R. Ladner. Improving generalization with active learning. *Machine learning*, 15(2): 201-221. 1994.
4. S. Duan, J. Zhang, P. Roe, and M. Towsey. A survey of tagging techniques for music, speech and environmental sound. *Artificial Intelligence Review*, 42(4): 637-661. 2014.
5. A. Fleury, N. Noury, M. Vacher, H. Glasson, and J. Seri. Sound and speech detection and classification in a health smart home. In *the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 4644-4647. 2008.
6. P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio, and M. Vento. Audio surveillance of roads: A system for detecting anomalous sounds. *IEEE Transactions on Intelligent Transportation Systems*, 17(1): 279-288. 2015.
7. A. Gadde, A. Anis, and A. Ortega. Active semi-supervised learning using sampling theory for graph signals. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 492-501. 2014.
8. S. Ghofrani, D. C. McLernon, and A. Ayatollahi. Comparing Gaussian and chirplet dictionaries for time-frequency analysis using matching pursuit decomposition. In *Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology*, pp. 713-716. 2003.
9. B. Gold, N. Morgan, and D. Ellis. Speech and audio signal processing: processing and perception of speech and music. *John Wiley & Sons*, 2011.
10. W. Han, E. Coutinho, H. Ruan, H. Li, B. Schuller, X. Yu, and X. Zhu. Semi-supervised active learning for sound classification in hybrid learning environments. *PloS One*, 11(9): 1-14. 2016.
11. A. Krogh, and J. Vedelsby. Neural network ensembles, cross validation, and active learning. In *Advances in Neural Information Processing Systems*, pp. 231-238. 1995.
12. C. Lei, and X. Zhu. Unsupervised feature selection via local structure learning and sparse learning. *Multimedia Tools and Applications*, 77(22): 29605-29622. 2018.
13. P. Maijala, S. Zhao, T. Heittola, and T. Virtanen. Environmental noise monitoring using source classification in sensors. *Applied Acoustics*, 129: 258-267. 2018.
14. S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12): 3397-3415. 1993.

15. D. Morrison, R. Wang, and L. C. De Silva. Ensemble methods for spoken emotion recognition in call-centres. *Speech Communication*, 49(2): 98-112. 2007.

16. H. Park, and C. Jun. A simple and fast algorithm for K-medoids clustering. *Expert Systems with Applications*, 36(2): 3336-3341. 2009.

17. N. C. Phuong, and T. D. Dat. Sound classification for event detection: Application into medical telemonitoring. In *International Conference on Computing, Management and Telecommunications*, pp. 330-333. 2013.

18. K. J. Piczak. ESC: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM International Conference on Multimedia*, pp. 1015-1018. 2015.

19. J. Ren, X. Jiang, J. Yuan, and N. Magnenat-Thalmann. Sound-event classification using robust texture features for robot hearing. *IEEE Transactions on Multimedia*, 19(3): 447-458. 2016.

20. G. Riccardi, and D. Hakkani-Tur. Active learning: Theory and applications to automatic speech recognition. *IEEE Transactions on Speech and Audio Processing*, 13(4): 504-511. 2005.

21. R. Rubinstein, A. M. Bruckstein, and M. Elad. Dictionaries for sparse representation modeling. *Proceedings of the IEEE*, 98(6): 1045-1057. 2010.

22. J. Salamon, C. Jacoby, and J. P. Bello. A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 1041-1044. 2014.

23. J. Schröder, J. Anemiiller, and S. Goetze. Classification of human cough signals using spectro-temporal Gabor filterbank features. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6455-6459. 2016.

24. R. V. Sharan, and T. J. Moir. Robust acoustic event classification using deep neural networks. *Information Sciences*, 396: 24-32. 2017.

25. S. Zhao, T. Heittola, and T. Virtanen. Active learning for sound event classification by clustering unlabeled data. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 751-755. 2017.

26. P. Sugden, and N. Canagarajah. Underdetermined noisy blind separation using dual matching pursuits. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 5: V-557. 2004.

27. P. Vera-Candeas, N. Ruiz-Reyes, M. Rosa-Zurera, D. Martinez-Munoz, and F. López-Ferreras. Transient modeling by matching pursuits with a wavelet dictionary for parametric audio coding. *IEEE Signal Processing Letters*, 11(3): 349-352. 2004.

28. R. Wang, and M. Zong. Joint self-representation and subspace learning for unsupervised feature selection. *World Wide Web*, 21(6): 1745-1758. 2018.

29. J. Wang, C. Lin, B. Chen, and M. Tsai. Gabor-based nonuniform scale-frequency map

for environmental sound classification in home automation. *IEEE Transactions on Automation Science and Engineering*, 11(2): 607-613. 2013.

30. C. Y. Wang, J. C. Wang, A. Santoso, C. C. Chiang, and C. H. Wu. Sound event recognition using auditory-receptive-field binary pattern and hierarchical-diving deep belief network. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(8): 1336-1351. 2017.

31. R. Wang, W. Ji, M. Liu, X. Wang, J. Weng, S. Deng, S. Gao, and C. Yuan. Review on mining data from multiple data sources. *Pattern Recognition Letters*, 109: 120-128. 2018.

32. Z. Zhang, and B. Schuller. Semi-supervised learning helps in sound event classification. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 333-336. 2012.

33. S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang. Efficient knn classification with different numbers of nearest neighbors. *IEEE Transactions on Neural Networks and Learning Systems*, 29(5): 1774-1785. 2017.

34. W. Zheng, X. Zhu, Y. Zhu, R. Hu, and C. Lei. Dynamic graph learning for spectral feature selection. *Multimedia Tools and Applications*, 77(22): 29739-29755. 2018.

35. X. J. Zhu. Semi-supervised learning literature survey. *University of Wisconsin-Madison Department of Computer Sciences*, 2005.

36. X. Zhu, S. Zhang, R. Hu, and Y. Zhu. Local and global structure preservation for robust unsupervised spectral feature selection. *IEEE Transactions on Knowledge and Data Engineering*, 30(3): 517-529. 2017.

37. T. Pranckevičius and V. Marcinkevičius. Comparison of naive Bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification. *Baltic Journal of Modern Computing*, 5(2): 221. 2017

38. Y. Dong, Y. Zhang, J. Yue, and Z. Hu. Comparison of random forest, random ferns and support vector machine for eye state classification. *Multimedia Tools and Applications*, 75(19): 11763-11783. 2016.

39. V., M. Sanchez-Castillo Rodriguez-Galiano, M. Chica-Olmo, and M. J. O. G. R. Chica-Rivas. Machine learning predictive models for mineral prospectively: An evaluation of neural networks, random forest, regression trees and support vector machines. *Ore Geology Reviews*, 71: 804-818. 2015.

40. A. Statnikov, L. Wang, and C. F. Aliferis. A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. *BMC bioinformatics*, 9(1): 319. 2008.

Chapter 3 A learnt dictionary-based active learning approach

Sound event tagging is a process that adds texts or labels to sound segments based on their salient features and/or annotations. In the real world, since the annotation cost is much expensive, tagged sound segments are limited, while untagged sound segments can be obtained easily and inexpensively. Thus, semi-automatic tagging becomes very important, which can assign labels to massive untagged sound segments according to a small number of manually annotated sound segments. Active learning is an effective technique to solve this problem, in which selected sound segments are manually tagged while other sound segments are automatically tagged. In this chapter, a learnt dictionary based active learning (LDAL) approach is proposed for environmental sound event tagging, which can significantly reduce the annotation cost in the process of semi-automatic tagging. The proposed approach is based on a learnt dictionary, as dictionary learning is more adaptable to sound feature extraction. Moreover, tagging accuracy and annotation cost are used to measure the performance of the proposed approach. Experimental results demonstrate that the proposed approach has higher tagging accuracy but requires much fewer annotation costs than other existing approaches.

This chapter is organized as follows. Section 3.1 introduces the background of sound event tagging and the motivation of this research. Section 3.2 reviews the most relevant works of this research. Section 3.3 presents the proposed sound event tagging approach in detail. Section 3.4 presents the experiments and discusses the experimental results. At the end of this chapter, the conclusion of the proposed approach is presented in Section 3.5.

3.1 Introduction

Tagging is a process that adds texts or labels to samples based on their salient features and/or annotations. Over the past few years, tagging is a fundamental challenge in the field of audio processing [18,19,20, 25, 35, 40] and image processing [30, 36, 39, 42]. Tagging approaches have been widely applied to the Internet of Things (IoT) [2], especially for object classification [28, 41], object detection [13, 29], monitoring [8, 16] and other real-world applications.

Environmental sounds refer to all non-verbal and non-communicatory sounds, including sounds produced by nature (such as weather and animals) and sounds created by human activities (such as traffic and indoor activities) [6, 9]. These sounds carry useful information and have their unique characteristics. In order to distinguish different environmental sound events and study them separately, sound event tagging is introduced. Recently, there are three types of tagging approaches: manual tagging approaches, automatic tagging approaches, and semi-automatic tagging approaches.

Manual tagging (*i.e.*, manually assign a predefined label to an untagged sound segment according to the result of annotation) provides an accurate and comprehensive audit of acoustic data. However, the annotation cost is much expensive, and the process of annotation is time-consuming, especially when the number of samples in a dataset is large. Different from manual tagging, automatic tagging can assign predefined labels to untagged sound segments automatically. However, the accuracy of automatic tagging is unsatisfactory in some cases, especially in acoustically complex environments.

Additionally, there are a limited number of tagged sound segments, whereas untagged sound segments are abundant in the real world. For example, the largest environmental sound dataset ESC-US [24] only contains a limited number of tagged sound segments (2000 recordings) and a large number of untagged sound segments (250,000 recordings). Facing the shortage of tagged sound segments, semi-automatic tagging approaches rise, which combines the advantages of both manual tagging and automatic tagging.

Semi-automatic tagging is a type of hybrid approach, which combines the advantages of both manual tagging and automatic tagging. Specifically, semi-supervised learning [24] is an effective technique for semi-automatic tagging. By using semi-supervised learning, untagged sound segments can be assigned labels according to the characteristics or distribution of tagged sound segments in the same dataset. However, these tagged sound segments which are used for semi-supervised learning based tagging approaches are preselected. Thus, in some cases, these preselected sound segments may not fully reflect the real distribution or characteristics of the whole dataset.

In order to address the above issue, active learning [3], as a special case of semi-supervised learning, is introduced to semi-automatic tagging approaches. Active learning can select untagged sound segments to be tagged from which it learns. The

sound segments, which are selected through active learning, will be more informative or more characteristic than the other sound segments in the same dataset. Thus, the selected untagged sound segments will be manually tagged. Then other untagged sound segments will be automatically tagged based on the similarities between the selected sound segments and themselves. In other words, if an informative sound segment is manually tagged first, other untagged sound segments that are much similar to this sound segment can be automatically tagged with the same label. Therefore, the annotation cost of sound event tagging approaches can be significantly reduced. This means active learning can achieve the maximum gain in the learning process with a small number of annotating requests [10, 31].

In recent years, dictionary learning has been widely used for environmental sound feature representation since sound features extracted based on dictionary learning have been proven to be able to perform the characteristics of environment sounds [5]. Multiple dictionary learning-based feature representation methods have been proposed, and the dictionaries in these methods can be divided into two categories: (i) predefined dictionary, in which the atoms (*i.e.*, feature vectors used to represent a signal) are preselected based on prior knowledge, such as Wavelets dictionary [21] and Gabor dictionary [17, 33], and (ii) learnt dictionary, which learns basis atoms from the signal itself. For environmental sound event tagging approaches, however, predefined dictionaries may not match the real structures of some environmental sounds, such as machinery sounds [10]. Thus, learnt dictionaries are introduced to learn better feature representations from environmental sounds. Efficient methods are designed to train learnt dictionaries, such as Maximum Likelihood (ML) [27], Method of Optimal Directions (MOD) [15], and K-SVD [7].

In this chapter, a learnt dictionary based active learning (LDAL) approach is proposed for environmental sound event tagging, which can significantly reduce the annotation cost in the process of semi-automatic tagging. The proposed approach combines dictionary learning for feature representation and active learning for semi-automatic tagging. Specifically, since learnt dictionary representations are suitable for describing the characteristics of environmental sounds, a single learnt dictionary is trained to extract features from untagged sound segments. Meanwhile, by using a *k*-medoids clustering method [1], the proposed approach will actively select the most informative sound segments to be annotated first (*i.e.*, assigned with their true labels). Then other untagged sound segments can be automatically tagged (*i.e.*, assigned with their predicted labels) according to the similarities between these selected sound segments and themselves. After that, a classifier, which is used to test the

performance of the proposed approach, will be trained using these tagged sound segments with their either true or predicted labels. In addition, sound event tagging accuracy and annotation cost are used to measure the experimental results of the proposed approaches.

The proposed approach is compared with several reference approaches on two public environmental sound datasets, named Environmental Sound Classification (ESC) dataset [24] and Urbansound8K dataset [14], respectively. Experimental results demonstrate that the proposed approach has the same or a higher tagging accuracy than other reference approaches but requires a much less annotation cost.

3.2 Related work

In recent years, many different approaches have been developed for sound event tagging. This section reviews the existing approaches for feature extraction and sound event classification, which are the two most relevant topics to our research.

Feature extraction is a dimensionality reduction process, through which the original features of data can be represented with more manageable representations. Sound event classification is a process of recognizing the set of active sound events in an audio segment based on the extracted acoustic features [43]. In this chapter, we focus on developing semi-supervised approaches to achieve sound event classification.

Previously, Huynh *et al*. [44] developed a Semi-Supervised Tree Support Vector Machine (SST-SVM) for cough recognition/classification, which required limited data for training. The developed SST-SVM was built based on Fisher Linear Discriminant (FLD) and could be retrained by using unlabeled test data with a confidence metric. However, this approach cannot perform well under noisy conditions.

To solve this problem, Terence *et al*. [34] proposed a robust adaptive semi-supervised Tree-SVM classifier for sound event classification. They extracted Mel-frequency Cepstral Coefficients (MFCCs) features from sound segments and adapted the extracted features using the proposed custom filter (*i.e*., a discriminative filter) constructed at each classification node of a tree. Compared with SST-SVM, this approach trained with limited data could achieve a higher discriminative capability (*i.e*., a higher classification accuracy), even under noisy conditions.

Further, Han *et al*. [11] developed an effective semi-supervised active learning approach for sound classification, which combined active learning and self-training to minimize the required annotation cost for sound classifier training. In their approach, a classifier confidence score was proposed to determine the informativeness of sound segments. If the confidence score of a sound segment was equal to or lower than the pre-defined threshold, the sound segment would be selected for human annotation; otherwise, it would be automatically labeled. However, the accuracy of this approach would be reduced under real-life noise conditions.

To solve this problem, Ye *et al*. [37] developed an aggregation approach, which combined both local and global acoustic features, for sound classification. A Mixture of Experts model (MoE) was utilized to formulate the relationships between local and global features and aggregated the heterogeneous acoustic information of sound segments for classification. However, their approach cannot extract features from variable-length sound segments.

Later, Jayalakshmi *et al*. [12] developed a feature extraction approach that could extract global statistical features from multi-variate varying length acoustic data. Then a discriminative model-based classifier was developed to detect acoustic events from audio segments. Their approach highly reduced the dimensionality of original acoustic features.

Recently, Zhao *et al*. [40] proposed an active learning-based approach for urban sound event classification. They extracted MFCC features from untagged sound segments and then developed an active learning approach using *k*-medoid clustering techniques. Thus, all sound segments could be manually and/or automatically labeled with the developed approach. However, since MFCCs perform better on structured sounds (such as speech and music) rather than on noise-like sounds (environmental sounds) [5], better feature extraction methods should be utilized or developed for environmental sound event classification.

In this chapter, a learnt dictionary-based active learning approach is developed for environmental sound event tagging, which combines dictionary learning for feature extraction and active learning for semi-automatic tagging. Figure 3.1 illustrates the procedure of the proposed approach. The performance of the proposed approach will be evaluated by the annotation cost and classification accuracy.

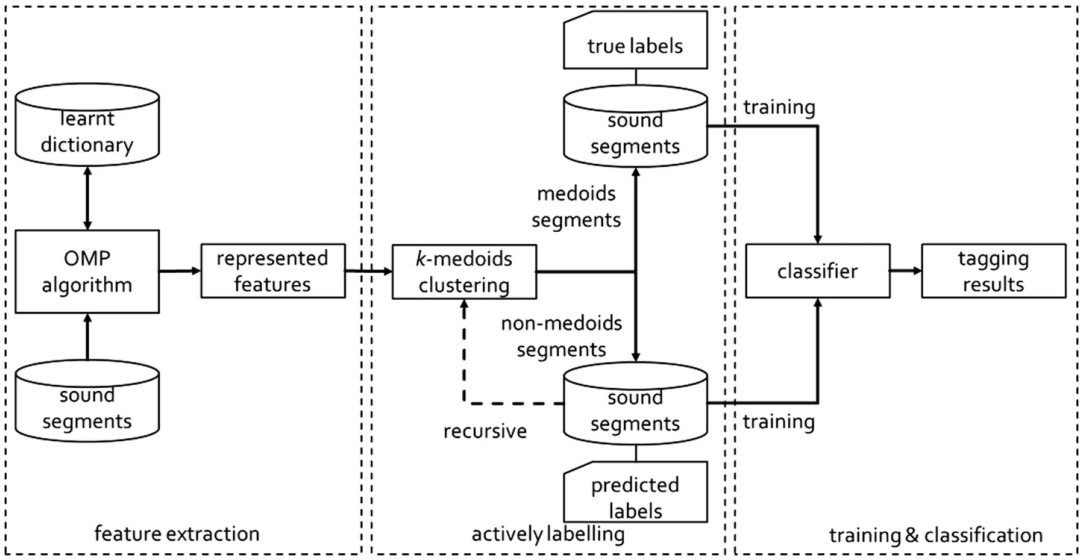


Figure 3.1. Flowchart of the proposed LDAL semi-automatic sound event tagging approach.

3.3 The proposed approach

As shown in Figure 3.1, the proposed learnt dictionary-based active learning approach consists of three phases: feature extraction, actively labeling, and sound event classification.

- In the feature extraction phase, a K-SVD based dictionary is utilized to extract features from unlabeled sound segments.

- In the actively labeling phase, a *k*-medoid clustering is used for sound segment selection based on the extracted features. According to the clustering results, the medoid of each cluster (*i.e.*, a sound segment) is considered as the most informative data point in its cluster. Thus, the medoid of each cluster will be annotated and be assigned with its true label, while other members in its cluster will be assigned the same label as their predicted labels. If more sound segments are required to be annotated, the *k*-medoid clustering process will be repeated. Thus, each sound segment can receive its true label and/or predicted labels according to the results of actively labeling.

- In the sound event classification phase, all sound segments with their true labels or predicted labels will be used to train a sound event classifier. The performance of the proposed approach is evaluated by classification accuracy. In addition,

sound event tagging accuracy and annotation cost are used to measure the performance of the proposed approach.

In the following sections, the details of each phase will be described respectively.

3.3.1 Feature extraction

In recent years, many different feature extraction approaches (*e.g.*, MFCCs, band energy ratio, zero crossing rates, and dictionary representation) were developed to extract features from environmental sounds. Compared to other feature extraction approaches, dictionary learning-based feature extraction approaches have the following advantages [5]:

- Dictionary learning-based approaches can extract time-frequency features rather than frequency features from sound segments.

- Dictionary learning-based approaches can perform better on noise-like sounds (*e.g.*, environmental sounds) while other feature extraction approaches such as MFCCs perform better on structured sounds (*e.g.*, speech and music).

Since the proposed approach utilizes a K-SVD learnt dictionary for feature extraction, the process of feature extraction can be divided into two sub-steps: (i) train a K-SVD learnt dictionary using untagged sound segments, (ii) represent untagged sound segments using the learnt dictionary.

3.3.1.1 Dictionary learning and sparse approximation

Let $X = \{x_i\}_{i=1,\dots,m}$ is a training set, where x_i is an input sound segment; $D \in \mathbb{R}^{\gamma \times t}$ represents a dictionary; $\alpha \in \mathbb{R}^t$ represents the sparse coefficient matrix of the input. The approximation of the input sound segments with N atoms can be formulated as:

$$\min_{D,\alpha}\{\|X - D\alpha\|_F^2\}, s.t. \|x_i\|_0 \leq N, \tag{3.1}$$

The dictionary matrix D, which is selected from a set of known transforms in a manual dictionary, will be trained by training samples in a learnt dictionary. Mathematically, a learnt dictionary can be generated by optimizing the following minimization problems [43]:

$$\min_{D,\{\alpha_i\}_{i=1,\ldots,m}} \sum_{i=1}^{m} \|x_i - D\alpha_i\|_F^2 + \mu\|\alpha_i\|, \quad (3.2)$$

where each $\{x_i\}_{i=1,\ldots,m}$ represents an input sound segment (*i.e.*, training samples), and μ is a penalty parameter that can balance the sparsity of the decomposition and the reconstruction error. The optimization problem in Equation (3.2) is usually not about the joint convexity of variables D and α.

To find a solution to the optimization problem in Equation (3.2), both sparse code α and dictionary D variable should be optimized. Either D or α needs to be fixed so that the objective function relative to the other variable can be changed to a convex function [23]. Therefore, the optimization problem will be solved in two steps:

- Sparse coding: Fixing the dictionary D, then the coefficients α of X will be calculated by minimizing Equation (3.2) solved through a pursuit algorithm.

- Dictionary update: To reduce the approximation error caused by applying K-SVD computation on the relevant samples, new dictionary D' will be calculated by using the obtained sparse coding matrix α.

These two steps work alternately and iteratively. The purpose of dictionary learning is to utilize as few atoms as possible to represent the input data in a given dictionary so that the information contained in the data can be obtained easily. The sparse coding step aims to find the sparsest representation with the least reconstruction error to represent the input sound segments. The dictionary update step is used to find basis vectors that can represent the input sound segments [23].

In the proposed approach, a standard K-SVD based dictionary is used rather than a variant of the standard K-SVD based dictionary for environmental sound feature extraction.

3.3.1.2 Matching pursuit algorithm

In the sparse coding step, the "best matching" projections of the training samples will be found by a pursuit algorithm. Efficient adaptive approximation techniques are developed such as Basis Pursuit (BP) [32], Matching Pursuit (MP) [4,17] and Orthogonal Matching Pursuits (OMP) [26], which represent data in the form of the linear combination of basis vectors (*i.e.*, atoms) from a dictionary. The OMP algorithm is introduced into the proposed approach, which decomposes the input

sound segments by using the atoms in an overcomplete dictionary and provided a sparse linear expansion of the input.

An OMP algorithm consists of two phases: selection and decomposition phases. In the selection phase, the OMP algorithm selects every atom from the current dictionary to check the close similarity between this atom and the input sound segments by computing the inner products between them.

Assume that dictionary D is a collection of atoms given by:

$$D = \{\phi_\gamma : \gamma \in \Gamma\}, \tag{3.3}$$

where Γ *denote*s the parameter set and ϕ_γ *denote*s an atom. Then the approximate decomposition of the input signal s can be represented as:

$$s = \sum_{i=1}^{n} \alpha_i \phi_{\gamma_i} + R^{(n)}, \tag{3.4}$$

where $R^{(n)}$ denotes the residual signal; α_i denotes the coefficient of $\phi_{\gamma i}$; n denotes the number of atoms used to represent s. Initially, the OMP algorithm calculates all inner products of s with the atoms in D. Comparing the similarity between an input sound segment and each atom by using their inner product, the atom with the largest absolute inner product $\phi_{\gamma 0}$ is selected as the first element. Mathematically, this can be represented as:

$$\left|\langle s, \phi_{\gamma_0}\rangle\right| \geq \left|\langle s, \phi_\gamma\rangle\right|, \forall \gamma \in \Gamma,$$

where <·> denotes the inner product operation and |·| denotes the absolute operation. Then the atom $\phi_{\gamma 0}$ is subtracted from s to get residual $R^{(0)}$. In this way, the approximation of s at the i^{th} step can be calculated by:

$$s^{(i)} = s^{(i-1)} + \alpha_i \phi_{\gamma_i}, \tag{3.5}$$

where $\alpha_i = \langle R^{(i-1)}, \phi_{\gamma_i}\rangle$ and $R^{(i)} = s - s^{(i)}$. After n steps, the stop criterion of the OMP algorithm is reached.

Different from the MP algorithm, the residual $R^{(i)}$ in the OMP algorithm is always

orthogonal to the span of the atoms already selected. This leads to better results of representation than the MP algorithm. Since the best atom is selected each time during the selection phase, the reconstruction error of the input signal will be minimal.

3.3.2 Actively labeling

In semi-automatic tagging approaches, untagged sound segments are assigned labels according to the characteristics or distribution of the tagged sound segments in the same dataset. However, these tagged sound segments are preselected and manually tagged. In some cases, these preselected sound segments may not fully reflect the real distribution or characteristics of the whole dataset.

Active learning can select untagged sound segments from which it learns. The sound segments, which are selected to be manually tagged through active learning, will be more informative or more characteristic than the other untagged sound segments in the dataset. Thus, the selected untagged sound segments will be manually tagged so that other untagged sound segments will be automatically tagged based on the similarities between untagged sound segments and tagged sound segments. In other words, if an informative sound segment is manually tagged, other untagged sound segments that are similar to this tagged sound segment can be automatically tagged with the same label. It means active learning can achieve the maximum gain in the learning process with a small number of annotation requests [10,31].

Our proposed approach utilizes active learning for actively labeling, the process of actively labeling consists of two phases: selection phase and labeling phase. Specifically, a *k*-medoids clustering algorithm is used for sample selection, then true labels or predicted labels will be actively assigned to these untagged sound segments according to the results of *k*-medoids clustering.

3.3.2.1 *K*-medoids clustering algorithm

Clustering is a processing that grouping a set of data objects into several subsets (*i.e.*, clusters). Data objects in the same cluster will be more similar to each other, but much different from the data objects in other clusters [22]. The *k*-medoids clustering algorithm is a centroid-based clustering process that iteratively finds *k* data objects as medoids and assigns other data objects to the nearest medoid of them. Since the clustering centroids of the *k*-medoids clustering algorithm are on *k* data objects, the

algorithm is less sensitive to outliers than *k*-means clustering algorithms [22]. The process of the *k*-medoids clustering algorithm is described in Algorithm 3.1.

The proposed approach utilizes the farthest-first traversal algorithm for initial medoid selection. This can effectively avoid local redundancy problems. A traversed set begins with a randomly selected sound segment. Euclidian distance is used as a dissimilarity measurement to calculate the distances between every pair of sound segment. Medoid update steps, *i.e.*, steps (3) and (4), calculate alternately and iteratively to minimize the accumulated distance between each medoid and the other member in its cluster until no medoid will be swapped to reduce the accumulated distance. The sound segment located at the centroid of a cluster is the medoid of the cluster.

Algorithm 3.1 *k*-medoids clustering algorithm

Input: number of clusters *k*, training object set $\alpha = \{\alpha_i\}_{i=1,\dots,m} \in \mathbb{R}^t$

Output: *k* cluster set $C = \{C_1, \dots, C_k\}$

(1) randomly select *k* objects as initial medoids $O = \{o_1, \dots, o_k\}$;

(2) calculate the Euclidian distance between every pair of objects;

(3) assign other objects to the nearest medoid in *O*;

(4) **for** each medoid o_* and the objects associated with o_*:

accumulate the dissimilarity distances between o_* and the objects associated with o_*;

find a new medoid $\tilde{o_*}$, the accumulated distance between the new medoid $\tilde{o_*}$ and other objects in the cluster are minimal;

update the current medoid o_* by replacing with the new medoid $\tilde{o_*}$;

(5) repeat step (3) and (4) alternately until there is no change in *O*;

(6) **return** *k* cluster set $C = \{C_1, \dots, C_k\}$;

(7) **end**

3.3.2.2 Actively labeling

The process of actively labeling is described as follows. Initially, all sound segments are untagged. After the process of *k*-medoids clustering, the medoid of each cluster can be considered as the most informative sample in its cluster. Due to the fact that the annotation cost is expensive, the number of *k* and annotation cost have the following relationships:

- If the annotation cost is less than k (*i.e.*, there are less than k samples can be annotated), all clusters will be sorted in descending order of size so that the medoid of a larger cluster will be annotated first.

- If the annotation cost is the same as k (*i.e.*, there are k samples can be annotated), all medoids will be annotated in order.

- If the annotation cost is larger than k (*i.e.*, there are more than k samples can be annotated), all medoids can be annotated, and then the k-medoids clustering will be processed repeatedly until the annotation cost is exhausted.

Meanwhile, while a medoid is tagged with its true label, other members in the same cluster will be assigned with labels, which are the same as the medoid's, as their predicted labels. The process of actively labeling is shown in Algorithm 3.2.

Algorithm 3.2 actively labeling algorithm

Input: annotation cost *sum*, number of cluster *k*, the represented training set $\alpha = \{\alpha_i\}_{i=1,\dots,m} \in \mathbb{R}^t$

Output: label set $L = \{l_1, \dots, l_m\}$

(1) **do** *k*-medoids clustering on the represented samples:
  if *sum* <= *k*:
    sort clusters in descending order;
    annotate the medoids of the largest *k* clusters;
    assign predicted labels to other samples;
  else
    annotate all medoids of *k* clusters;
    assign predicted labels to other samples;
    sum = *sum* – *k*;
    repeat step (1);
  end

(2) **return** *L*

(3) **end**

The purpose of the proposed approach is to tag environmental sound segments with the least annotation cost rather than to achieve a higher tagging accuracy by annotating more sound segments (*i.e.*, assigning more true labels). Thus, larger clusters in k clusters will be tagged first. When the medoid of the largest cluster is tagged, the largest number of predicted labels can be assigned to the other untagged

sound segments. In other words, this true label can be spread farthest. According to the relationships between the number of *k* and the annotation cost, a sound segment can have a true label, at least one predicted label, or do not have any predicted labels.

3.3.3 Sound event classification

With the proposed approach, a classifier, which is used to distinguish different environmental sound events, is trained using all training sound segments with their either true or predicted labels. The accuracy of the trained classifier is used to evaluate the performance of our proposed approach.

The main idea of the proposed approach is to assign labels to untagged sound segments according to a small number of manually annotated sound segments. Through active learning, the proposed approach significantly reduces the annotation cost in the process of tagging. In other words, the proposed approach aims to reduce the annotation cost rather than to improve the classification accuracy. Thus, a standard classifier is trained to evaluate the tagging results.

Algorithm 3.3 LDAL for sound event tagging

Input: annotation cost *sum*, number of cluster *k*,
training set $x = \{x_i\}_{i=1,\dots,m}$, text set $x' = \{x'_j\}_{j=1,\dots,m}$

Output: predicted labels for the test set L'

(1) set the dictionary matrix $D^{(0)} \in \mathbb{R}^{\gamma \times t}$ with l_2 normalized columns;

(2) repeat until convergence (stop rule):
  sparse coding;
  update dictionary;

(3) use dictionary *D* to calculate the sparse coefficient matrix α to represent training set *x*;

(4) do *k*-medoids clustering on the represented samples α;

(5) assign labels based on the relationships between *k* and *sum*;

(6) repeat step (4) and (5) until the annotation cost is exhausted;

(7) get label set *L* of the training samples;

(8) train a classifier using the training samples *x* with their labels in *L*;

(9) **end**

Algorithm 3.3 shows the whole process of the proposed approach. Specifically, steps (1)-(3) are used to learn a dictionary and represent sound segments using the learnt

dictionary, steps (4)-(6) are used to tag sound segments based on active learning, steps (7) and (8) are used to train a classifier for sound event tagging using the input sound segments and their either true or predicted labels.

3.4 Experiments

To measure the performance of the proposed approach, two environmental sound datasets are used, named Urbansound8K [26] and ESC [24], respectively.

3.4.1 Datasets and experiment setup

The UrbanSound8K dataset [26] is a public urban sound dataset that includes 8732 tagged sound segments (<=4s) of urban sounds from 10 classes (*i.e.*, air_conditioner, car_horn, children_playing, dog_bark, drilling, enginge_idling, gun_shot, jackhammer, siren, and street_music). All sound segments are pre-sorted into ten folds for cross-validates the automatic classification results.

The ESC dataset [24] is a collection of short environmental sound segments that includes 50 classes. In this chapter, a subset ESC-10 with 10 classes (*i.e.*, dog bark, rooster, rain, sea waves, crackling fire, crying baby, sneezing, clock tick, helicopter, and chainsaw) is used for all experiments. All sound segments are presorted into ten folds for cross-validates the automatic classification results.

In our experiments, each environmental sound segment in the dataset makes up an instance for training or testing. Each sound segment is divided by a frame window of 512 points with a 50% overlap. Using the K-SVD algorithm, a learnt dictionary is trained using the training set. The number of atoms selected from each learnt dictionary is tested repeatedly, 40 atoms are selected for feature representation in the UrbanSound8K dataset, and 10 atoms are selected for feature representation in the ESC-10 dataset. The following summary statistics of the represented samples are used as segment-wise features: maximum, minimum, medium, mean, and standard deviation.

In each round of evaluation, 90% of sound segments in the given dataset are used for training, while 10% of sound segments are used for testing. The final results are reported as the average of these ten results. All the experiments are written in Python 2.7 and processed on an HP Elite Desk 800 workstation with Intel i7–4790 CPU and 16GB RAM. The labels provided by the given dataset are used as ground truth. In a

training set, all true labels are hidden initially. In addition, the annotation cost *sum* is the number of true labels assigned to the dataset.

A supervised multi-class Support Vector Machine (SVM) classifier is used as the sound event classifier. The classifier is trained by the environmental sound segments in the training set with their either true or predicted labels. Since the proposed approach does not aim to optimize the classification model or SVM, the parameters of this classifier come from the default setting of the SVM classifier in Python Scikit-learn (http://scikit-learn.org/stable/index.html). We also test other classification models such as random forest classifiers and decision tree classifiers in our experiments with their default setting in Python Scikit-learn. Since the classifiers utilize in reference approaches are SVM, the classification results of SVM classifier are presented as the experimental results. All experiments are repeated ten times and the averaged results are reported.

3.4.2 Reference approaches

To evaluate the performance of the proposed approach, four semi-supervised learning or active learning-based approaches and our proposed approach are tested on the given two datasets respectively.

Semi-supervised learning (SSL) for sound event tagging is used as the first reference approach. Random sampling, which can be used to simulate the performance of semi-supervised learning, is used to select samples to be manually tagged. The annotation cost is the number of preselected samples [40].

Certainty based active learning (CRTAL) [38], as the second reference approach, is an active learning approach for speech recognition. In CRTAL, the annotation cost will be divided into two parts, one is for sample selection (*i.e.*, randomly choosing samples to be manually tagged), and the other is for uncertainty selection (*i.e.*, checking the samples with high uncertainty). The batch size is set as five. In each iteration, five samples, which have the least confidence in the current framework, will be tagged. Then the framework will be updated by adding new labels to the training sets.

The third reference approach is a medoid based active learning approach (MAL) [40]. All samples are represented with summary statistics of MFCCs, including minimum, maximum, median, mean, variance, skewness, kurtosis, median, and the variance of

the first and second derivatives. In MAL, active learning is utilized for sample selection.

The last reference approach is the DBAL approach proposed in Chapter 2.

3.4.3 Experimental results

Figure 3.2 shows the performance of the proposed sound event tagging approach (*i.e.*, learnt dictionary-based active learning approach (LDAL)) compared with the reference approaches on the UrbanSound8K dataset. With the increasing annotation cost, the lines which are used to present classification accuracy raise nonlinearly. Assuming that the predicted labels assigned to the samples are consistent with the true labels of these samples, the accuracy of the proposed classifier, which is trained using training samples with their either true or predicted labels, can achieve 67.5%.

Figure 3.2 also demonstrates that dictionary learning-based feature representation can improve the performance of the final results. Since SSL is a semi-supervised approach, which randomly selects recordings to be manually labeled, it achieves the lowest classification accuracy. CRTAL and MAL are two active learning-based approaches, the classification accuracies of these two approaches show a similar increasing trend as the annotation cost increases. Compared with MAL, DBAL achieves higher classification accuracy since it utilizes a Gabor dictionary for feature extraction. Our LDAL approach achieves the highest classification accuracy since the leant dictionary extracts more effective features from sound recordings than the Gabor dictionary. As shown in the experimental results, when the annotation cost is 3000, other reference approaches (*i.e.*, SSL, CRTAL, and MAL) can achieve an acceptable classification accuracy, while our approach achieves similar accuracy using only half of the annotation cost. When the annotation cost is 8000 (*i.e.*, every sound segment in the training set is assigned with its true label), the accuracy of the proposed approach is higher than that of the reference approaches (*i.e.*, SSL, CRTAL, and MAL), in which the features of the training set are not extracted by using dictionary learning.

Figure 3.3 shows the classification accuracy in the ESC dataset by using the proposed approach and reference approaches when annotation cost is set to 100 (about one-third of the total training samples). As the number of annotation cost increases, classification accuracy will increase gradually. Thus, we set annotation cost is 100, an appropriate annotation cost, to verify the accuracy of each category when the

annotation cost is fixed. Compared with other reference approaches, our proposed approach achieved the highest classification accuracy in classifying sounds ‘dog bark’, ‘crying baby’, and ‘clock tick’. In addition to the several reference approaches mentioned above, two supervised learning-based classification approaches, in which the classifiers are trained by using all training samples with their true labels, are proposed for comparison (the classification results are provided by the ESC dataset in [24]). The last column ‘average’ is the average classification accuracy of all categories.

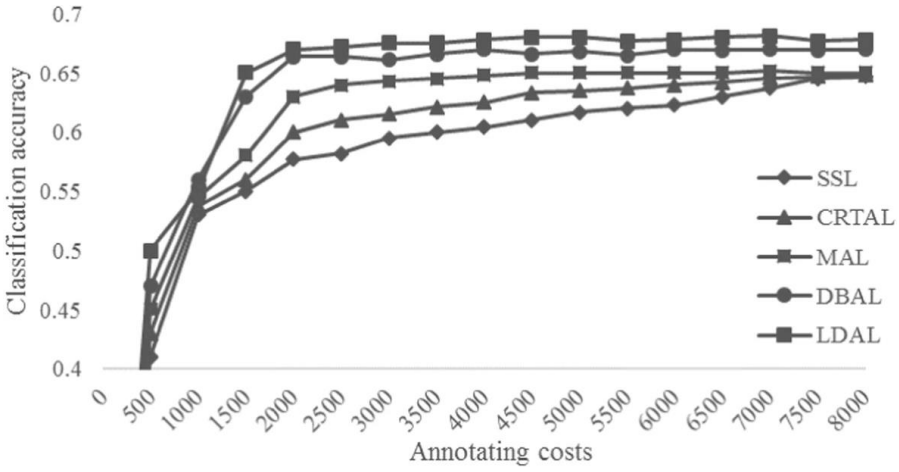


Figure 3.2. Classification accuracy on the UrbanSound8K dataset.

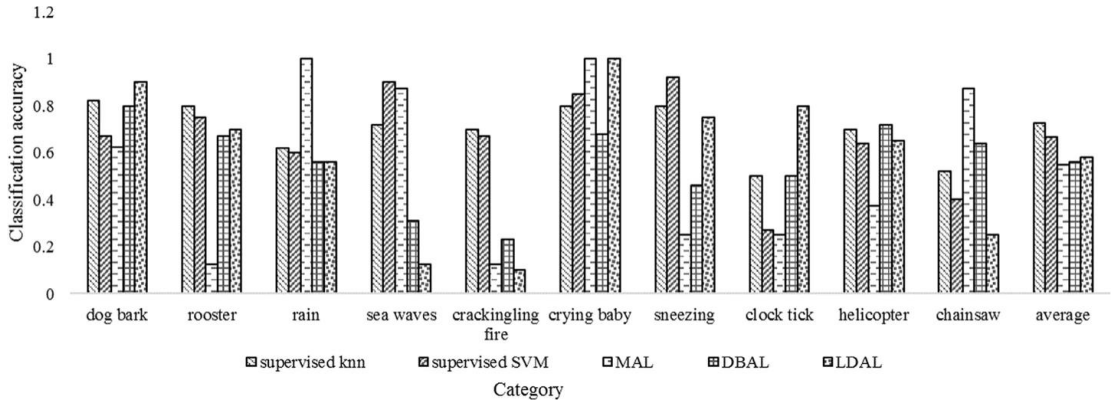


Figure 3.3. Classification accuracy on the ESC-10 dataset.

3.5 Summary

In this chapter, a learnt dictionary-based active learning approach is proposed for environmental sound event tagging, which can significantly reduce the annotation cost in the process of semi-automatic tagging. Specifically, a learnt dictionary is

utilized to extract features from environmental sound segments. To save annotation cost, active learning is employed for sample selection, in which environmental sound segments will be manually tagged to get true labels or automatically tagged to get predicted labels selectively. Moreover, a multi-class classifier, which is trained using sound segments with their either true or predicted labels, is trained to measure the proposed approach. The sound event tagging accuracy and annotation cost are used to measure the performance of the proposed approach. Experimental results demonstrate that the proposed approach has higher tagging accuracy but requires much less annotation cost than other existing approaches.

The proposed approach and reference approaches are tested on the UrbanSound8K dataset and the ESC dataset. Both annotation cost and classification accuracy are used for evaluation. Experimental results demonstrate that the proposed approach received a classification rate of 67.5% in unknown environments, without any preprocessing or prior knowledge. According to the experimental comparisons, the proposed approach can achieve a higher tagging accuracy but requires less annotation cost than reference approaches.

References

1. M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing*, 54(11): 4311-4322. 2006.
2. B. L. R. Stojkoska, and K. V. Trivodaliev. A review of Internet of Things for smart home: Challenges and solutions. *Journal of Cleaner Production*, 140: 1454-1464. 2017.
3. O. Chapelle, B. Scholkopf, and A. Zien. Semi-supervised learning. *IEEE Transactions on Neural Networks*, 20(3): 542-542. 2009.
4. S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Review*, 43(1): 129-159. 2001.
5. S. Chu, S. Narayanan, and C. C. J. Kuo. Environmental sound recognition with time-frequency audio features. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(6): 1142-1158. 2009.
6. S. Duan, J. Zhang, P. Roe, and M. Towsey. A survey of tagging techniques for music, speech and environmental sound. *Artificial Intelligence Review*, 42(4): 637-661. 2014.
7. K. Engan, S. O. Aase, and J. H. Husoy. Method of optimal directions for frame design. In *IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings*, pp. 2443-2446. 1999.

8. A. Fleury, N. Noury, M. Vacher, H. Glasson, and J. Seri. Sound and speech detection and classification in a health smart home. In *the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 4644-4647. 2008.

9. P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio, and M. Vento. Audio surveillance of roads: A system for detecting anomalous sounds. *IEEE Transactions on Intelligent Transportation Systems*, 17(1): 279-288. 2015.

10. A. Gadde, A. Anis, and A. Ortega. Active semi-supervised learning using sampling theory for graph signals. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 492-501. 2014.

11. W. Han, E. Coutinho, H. Ruan, H. Li, B. Schuller, X. Yu, and X. Zhu. Semi-supervised active learning for sound classification in hybrid learning environments. *PloS One*, 11(9): 1-14. 2016.

12. S. L. Jayalakshmi, S. Chandrakala, and R. Nedunchelian. Global statistical features-based approach for acoustic event detection. *Applied Acoustics*, 139: 113-118. 2018.

13. W. Ji, R. Wang, and J. Ma. Dictionary-based active learning for sound event classification. *Multimedia Tools and Applications*, 78(3): 3831-3842. 2019.

14. X. Jin, and J. Han. K-medoids clustering. *Encyclopedia of Machine Learning*. pp. 564-565. 2010.

15. M. S. Lewicki, and T. J. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12(2): 337-365. 2000.

16. P. Maijala, S. Zhao, T. Heittola, and T. Virtanen. Environmental noise monitoring using source classification in sensors. *Applied Acoustics*, 129: 258-267. 2018.

17. S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12): 3397-3415. 1993.

18. A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen. Acoustic event detection in real life recordings. In *the 18th European Signal Processing Conference*, pp. 1267-1271. 2010.

19. D. Morrison, R. Wang, and L. C. D. Silva. Spoken affect classification using neural networks. In *IEEE International Conference on Granular Computing*, pp. 583-586. 2005.

20. D. Morrison, R. Wang, L. C. D. Silva, and W. L. Xu. Real-time spoken affect classification and its application in call-centres. In *the 3rd International Conference on Information Technology and Applications*, pp. 483-487. 2005.

21. B. Ophir, M. Lustig, and M. Elad. Multi-scale dictionary learning using wavelets. *IEEE Journal of Selected Topics in Signal Processing*, 5(5): 1014-1024. 2011.

22. H. Park, and C. Jun. A simple and fast algorithm for K-medoids clustering. *Expert Systems with Applications*, 36(2): 3336-3341. 2009.

23. Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In

Proceedings of 27th Asilomar Conference on Signals, Systems and Computers, pp. 40-44. 1993.

24. K. J. Piczak. ESC: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM International Conference on Multimedia*, pp. 1015-1018. 2015.
25. J. Ren, X. Jiang, J. Yuan, and N. Magnenat-Thalmann. Sound-event classification using robust texture features for robot hearing. *IEEE Transactions on Multimedia*, 19(3): 447-458. 2016.
26. J. Salamon, C. Jacoby, and J. P. Bello. A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 1041-1044. 2014.
27. J. Schröder, J. Anemiiller, and S. Goetze. Classification of human cough signals using spectro-temporal Gabor filterbank features. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6455-6459. 2016.
28. R. V. Sharan, and T. J. Moir. Robust acoustic event classification using deep neural networks. *Information Sciences*, 396: 24-32. 2017.
29. J. Shen, Z. Chen, C. Xu, and H. Wang. Polarization and solar altitude correlation analysis and application in object detection. In *International Conference on Progress in Informatics and Computing*, pp. 179-183. 2017.
30. Y. Shi, Y. Gao, R. Wang, Y. Zhang, and D. Wang. Transductive cost-sensitive lung cancer image classification. *Applied Intelligence*, 38(1): 16-28. 2013.
31. S. Tong, and D. Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2(11): 45-66. 2001.
32. G. Tüysüzoğlu, and Y. Yaslan. Sparse coding based classifier ensembles in supervised and active learning scenarios for data classification. *Expert Systems with Applications*, 91: 364-373. 2018.
33. P. Vera-Candeas, N. Ruiz-Reyes, M. Rosa-Zurera, D. Martinez-Munoz, and F. López-Ferreras. Transient modeling by matching pursuits with a wavelet dictionary for parametric audio coding. *IEEE Signal Processing Letters*, 11(3): 349-352. 2004.
34. N. W. Z. Terence, T. H. Dat, H. T. Hoa, and C. E. Siong. Adaptive semi-supervised tree SVM for sound event recognition in home environments. In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pp. 1-4. 2013.
35. C. Y. Wang, J. C. Wang, A. Santoso, C. C. Chiang, and C. H. Wu. Sound event recognition using auditory-receptive-field binary pattern and hierarchical-diving deep belief network. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(8): 1336-1351. 2017.
36. R. Wang, W. Ji, M. Liu, X. Wang, J. Weng, S. Deng, S. Gao, and C. Yuan. Review on mining data from multiple data sources. *Pattern Recognition Letters*, 109: 120-128. 2018.

37. J. Ye, T. Kobayashi, and M. Murakawa. Urban sound event classification based on local and global features aggregation. *Applied Acoustics*, 117: 246-256. 2017.

38. J. Zhang, and H. Yuan. A certainty-based active learning framework of meeting speech summarization. In *Computer Engineering and Networking*, pp. 235-242. 2014.

39. S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang. Efficient knn classification with different numbers of nearest neighbors. *IEEE Transactions on Neural Networks and Learning Systems*, 29(5): 1774-1785. 2017.

40. S. Zhao, T. Heittola, and T. Virtanen. Active learning for sound event classification by clustering unlabeled data. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 751-755. 2017.

41. S. Zhao, T. Heittola, and T.s Virtanen. Learning vocal mode classifiers from heterogeneous data sources. In *IEEE Workshop Applications of Signal Processing to Audio and Acoustics*, pp. 16-20. 2017.

42. X. Zhu, S. Zhang, R. Hu, and Y. Zhu. Local and global structure preservation for robust unsupervised spectral feature selection. *IEEE Transactions on Knowledge and Data Engineering*, 30(3): 517-529. 2017.

43. H. Fayek, V. Tourbabin, and S. Adavanne. Sound event classification and detection with weakly labeled data. In *Detection and Classification of Acoustic Scenes and Events*, pp. 15-19. 2019.

44. T. H. Huynh, V. A. Tran, and H. D. Tran. Semi-supervised tree support vector machine for online cough recognition. In *Annual Conference of the International Speech Communication Association*, pp. 1637-1640. 2011.

Chapter 4 An attention-based dual learning approach

Video captioning aims to generate sentences/captions to describe video content. It is a key task in the field of digital video processing. However, most existing video captioning approaches only utilized the visual information of the video to generate captions. Recently, a new encoder-decoder-reconstructor architecture was developed for video captioning, which used the information in both raw videos and generated captions to generate video captions through dual learning. Based on this architecture, this chapter proposes a novel attention based dual learning approach (ADL) for video captioning. Specifically, ADL consists of two modules, a caption generation module and a video reconstruction module. The caption generation module builds a translatable mapping between raw video frames and generated video captions, *i.e.*, using the visual features extracted from videos by an Inception-V4 network to produce video captions. The video reconstruction module reproduces the raw video frames using the generated video captions, *i.e.*, using the hidden states of the decoder in the caption generation module to reproduce/synthesize the raw visual features. A multi-head attention mechanism is used to help the two modules attend to the most effective information in videos and captions, and a dual learning mechanism is used to fine-tune the two modules. Therefore, the proposed approach can minimize the semantic gap between videos and generated captions by minimizing the differences between the reproduced videos and the raw videos, thereby improving the quality of the generated video captions. Experimental results demonstrated that the proposed video captioning approach is superior to the state-of-the-art approaches on benchmark datasets.

This chapter is organized as follows. Section 4.1 introduces the background of video captioning and the motivation of this research. Section 4.2 reviews the most relevant works of this research. Section 4.3 presents the proposed approach in detail. Section 4.4 presents the experiments and discusses the experimental results. At the end of this chapter, the conclusion of the proposed approach is presented in Section 4.5.

4.1 Introduction

Video captioning aims to generate sentences/captions that can describe video content [1-3]. It has received increasing attention in the fields of video understanding [4],

natural language processing [5,6], and computer vision [6-8]. In the real world, video captioning based applications, such as video captioning based transcriptions and blind navigation, are widely used in surveillance systems, healthcare, and smart cities, and demonstrate their enormous scientific and commercial potential in these applications [4].

Compared to other captioning tasks (*e.g.*, image captioning [9-11]), video captioning is more challenging. This is because a video contains more complicated information (*e.g.*, actions, objects, and scenes) than a still image [19]. The existing video captioning approaches are mainly based on two types of models: (i) template-based language models and (ii) sequence learning-based models.

Early efforts on video captioning mainly focused on template-based language models. The template-based language models [11-14] predefined a series of language templates and mapped video features to words using object detection methods. Then the detected words would be placed on a predefined template to form a video caption that followed specific grammatical rules to describe the video content. Thus, each part of the generated sentence could be aligned with the video content based on the predefined templates. However, since the captions were composed of the detected words, template-based language models only described the detected video contents, *i.e.*, part of the video contents. Furthermore, since the syntactical structure of a caption was predefined by the templates, the generated caption was kind of 'robotic', *i.e.*, not quite like a natural language sentence [19].

Recently, various deep learning techniques have obtained significant success in the fields of image processing and speech processing [15-18]. These techniques have also been introduced to the video captioning task. The video captioning models based on these deep learning techniques are named sequence learning based models, also known as the encoder-decoder models [19].

A sequence learning-based model usually includes two stages: encoding stage and decoding stage. In the encoding stage, convolutional neural networks (CNNs) are often used as an encoder to convert a video into a compact vector to extract video features from videos. After that, the extracted video features are fed into a recurrent neural network (RNN) based decoder for generating video captions. Compared with the video captions generated by the template-based language models, the video captions generated by the sequence-based learning models have more flexible syntactical structures.

Furthermore, since the encoder-decoder models allow the training process to work in an end-to-end manner, they have become the mainstream of current video captioning models. However, the encoder-decoder models have a limitation in generating video captions. Specifically, in the training process of an encoder-decoder model, the previous ground-truth word is often used as the input of the decoder at each time step. But, in the test process, the input of the decoder is replaced by the previously generated word that is generated by the decoder [20]. This exposure bias may lead to error accumulation during the test process. In other words, during the test process, once a "bad" word is generated by the model, this error will be propagated and accumulated as the length of the sequence increases.

To overcome the aforementioned problem, a reconstruction network (RecNet) was proposed in [1] with a new encoder-decoder-reconstructor architecture. The proposed network generated video captions through the dual learning on two flows (a video-to-sentence flow and a sentence-to-video flow). Specifically, the video-to-sentence flow encoded video semantic features to produce video captions, and the sentence-to-video flow reconstructed the video features using the output of the video-to-sentence flow. A soft attention mechanism was used in both flows to capture key information from video features and generated captions. However, this simple temporal attention mechanism cannot capture the internal relationships between various key information [40].

To overcome the above problem, this chapter proposes a novel attention-based dual learning approach (ADL) for video captioning. Based on the similar architecture in [1], a multi-head attention mechanism replaces the soft attention mechanism to capture key information from raw videos and generated video captions. Specifically, two modules (*i.e.*, a caption generation module and a video reconstruction module) are contained in the proposed approach. The caption generation module is developed to generate video captions using the visual features extracted from videos through an Inception-V4 network. The video reconstruction module reproduces/synthesizes the raw video feature sequences (*i.e.*, the raw video frames) using the hidden states of the decoder in the caption generation module. The multi-head attention mechanism is used in the two modules to help them focus on the most effective information in raw videos and video captions, and a dual learning mechanism is used to fine-tune the two modules. Therefore, the proposed approach can minimize the semantic gap between raw videos and generated captions by minimizing the differences between the reproduced and the raw videos, thereby enhancing the quality of the generated video captions. Experimental results demonstrated that our approach is superior to

the state-of-the-art approaches for video captioning on benchmark datasets.

4.2 Related work

Video captioning has received extensive attention in the fields of video understanding, natural language processing, and computer vision in recent years. The existing video captioning approaches can be classified into two categories: template-based language models and sequence learning-based models. In this section, Sections 4.2.1 and 4.2.2 briefly introduce the two types of video captioning models, and then Section 4.2.3 reviews the applications of dual learning.

4.2.1 Template-based language models

Early work [11-14] for video captioning mainly relied on template-based language models, which predefined a set of language templates for caption generation. Specifically, a sentence was separated into several phases (*e.g.*, subject, verb, and object) based on the predefined templates with specific grammar rules [19]. By using object detection methods, each word was aligned with a part of video information, and then all detected words were placed in different phases of a template to generate a video caption. To detect objects in a video, Kojima *et al*. [12] developed a human activity description method based on concept hierarchies of actions. However, the generation of their approach was limited to narrow domains and small vocabularies of actions. To describe arbitrary activities in videos, Guadarrama *et al*. [8] developed an approach named zero-shot recognition to recognize activities in a video and described the recognized activities using semantic hierarchies.

Different from the above works, Rohrbach *et al*. [13] developed a video captioning approach that introduced a conditional random field (CRF) to simulate the connections/relationships between objects and activities in a video. Thus, in their approach, both visual features and semantic features were used for generating video captions. Further, Xu *et al*. [14] developed a video captioning framework that contained a joint embedding module, a deep video module, and a semantic language module, to generate video captions from videos.

However, since the template-based language models were incapable of textualizing everything in videos, *i.e*., mapping all video information to words, the sentences generated by these models only described part of video contents. In addition, since

the templates predefined the syntactic structures of video captions, the generated sentences were based on simple and uniform syntactic structures, which were kind of robotic in some cases. Thus, the sequence learning-based model has been developed for video captioning.

4.2.2 Sequence learning-based models

Recent achievements in deep learning techniques have significantly enhanced the performance of video captioning approaches. Compared to template-based language models, sequence learning based models can generate video captions with more flexible syntactical structures. This is because the model can learn the probability distributions of video contents and natural language sentences in a common space.

A typical architecture of the sequence learning-based video captioning model is to combine CNNs with RNNs, where CNNs are utilized to extract compact representational vectors from the input videos, and RNNs are utilized to construct a language model that operates on the extracted vectors for video caption generation. Venugopalan *et al*. [21] computed video representation vectors by averaging the features of each video frame extracted by CNNs, and then these vectors were fed into a Long Short Term Memory network (LSTM) for caption generation.

To capture the temporal dynamics of video sequences, Venugopalan *et al*. [22] developed the well-known Sequence to Sequence Video to Text (S2VT) approach, which utilized the optical flow to extract temporal information, and used LSTMs on both the encoder and the decoder. Zhang and Tian [23] proposed a two-stream neural network to exploit both spatial and temporal information for video captioning.

Furthermore, attention mechanisms were introduced to the video captioning models, which have been proven as an effective way to enhance the performance of video captioning models with the encoder-decoder structure. Yao *et al*. [24] proposed a temporal attention mechanism to exploit the global temporal structure of videos. The proposed attention mechanism could assign weights to video frame features, and the weighted frame features were used to generate video captions. Yan *et al*. [3] proposed a spatial-temporal attention mechanism (STAT) for video captioning, which captured information from the spatial-temporal structures in a video and selected the significant regions from the most relevant video segments to generate captions. However, this approach only considered visual information for caption generation.

Recently, since a video contains multiple modalities, such as visual modality, audio modality, and textual modality, multimodal learning was also introduced to video captioning models to improve the quality of the generated captions. Wang *et al*. [2] proposed a Multimodal Memory Model (M3) for video captioning based on textual and visual modalities to solve the visual-textual alignment problem. They developed a visual and textual shared memory that modeled long-term visual-textual dependency and guided visual attention for video caption generation by interacting videos and captions.

4.2.3 Dual learning approaches

Dual learning has been effectively applied for many machine learning applications, such as machine translation [25-28], image-to-image transformation [29-31], sentiment analysis [32], image segmentation [33], *etc*. The main idea of dual learning is very intuitive: leveraging the duality between two related tasks as a feedback signal to boost the performances of both tasks [34,35].

Usually, a dual learning framework contains two agents (a primal model and a dual model) to utilize such duality. The primal model maps an object x from one domain to another, while the dual model map it back. The mapping functions between these two domains are trained simultaneously so that one function is close to the inverse of the other. For example, when using dual learning in machine translation, if we translate a sentence from Chinese to English and then translate the obtained English sentence back to Chinese, the same sentence or a very similar one can be obtained.

He *et al*. [25] first proposed dual learning and applied it to machine translation. They updated two dual translators in a reinforcement learning manner and utilized the reconstructed distortion as the feedback signal. After that, Wang *et al*. [26] and Xia *et al*. [36] considered the joint distribution constraint in dual learning. They have proved that the joint distribution of samples over two domains is invariant when computing from either domain. Xia *et al*. [37] proposed a model-level dual learning approach, which shared components between the primary model and the dual model.

In addition, Zhao *et al*. [20] proposed a cross-domain image captioning approach using dual learning to overcome the problem of lack of image-text pairs in the training set. Wang *et al*. [34] proposed a multi-agent dual learning framework, which consisted of multiple primal and dual models, for machine translation and image translation.

In this chapter, our proposed approach utilizes attention based dual learning for video captioning. Unlike the existing encoder-decoder model which only contains a video-to-caption forward flow, we also build a caption-to-video backward flow. In other words, by fully considering the bidirectional training between videos and captions, our proposed approach is able to further enhance the accuracy of video captioning.

4.3 The proposed approach

This chapter proposes a novel attention based dual learning approach for video captioning. As illustrated in Figure 4.1, ADL includes two modules: a caption generation module and a video reconstruction module. The caption generation module constructs the forward flow from videos to captions by learning a translatable mapping between video frames and captions. The backward flow from captions to videos is formed by the video reconstruction module, which is able to synthesize raw video feature sequences based on the hidden state sequences of the decoder. A multi-head attention mechanism is used in the two modules, helping them focus on the most effective information for video captioning. The two modules are fine-tuned via dual learning, and the whole approach is trained in an end-to-end fashion.

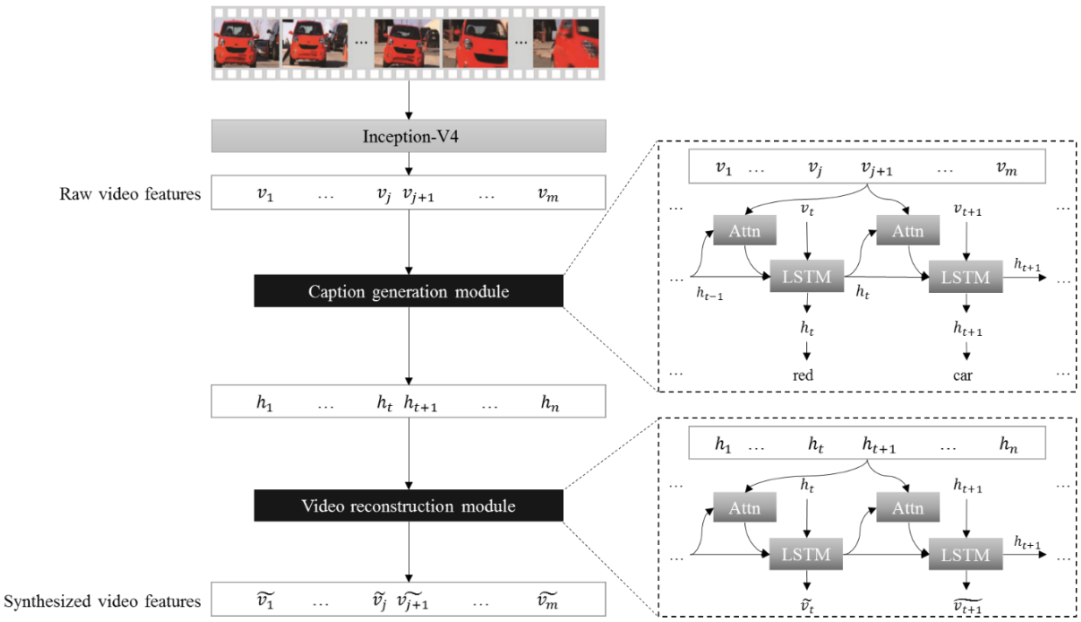


Figure 4.1. Illustration of the proposed attention-based dual learning approach for video captioning.

In this section, a brief introduction of RNN and LSTM is provided in Section 4.3.1, and the two modules are presented in Sections 4.3.2 and 4.3.3, respectively. The loss

function of the proposed approach is presented in Section 4.3.4 for training.

4.3.1 Long short-term memory recurrent neural network

The recurrent neural network is a class of deep neural networks extended from the feedforward neural network by adding feedback connections. RNNs have shown extraordinary capability in dealing with sequence learning. It is because it contains a specially designed recurrent operation that models sequence information by maintaining the historical sequential information inside hidden units.

Specifically, given a sequence of input vectors $\{x_1, x_2, \ldots, x_n\}$, a standard RNN can calculate the output sequence $\{y_1, y_2, \ldots, y_T\}$ according to the following equations:

$$h_t = \phi(W_h x_t + U_h h_{t-1} + b_h), \tag{4.1}$$

$$y_t = \phi\big(U_h h_t + b_y\big), \tag{4.2}$$

where $\phi(\cdot)$ denotes an activation function; h_t denotes the hidden state at time step t ($t = 1, \ldots, T$); matrices W_* and U_* denote the weights to be learned; b_* denotes a bias term. Thus, the input x_t and the previous hidden layer's state h_{t-1} can be utilized to obtain the current hidden layer state h_t and current hypothesis y_t. The historical information of a sequence is transmitted throughout the whole sequence and affects the output at each time step.

However, standard RNNs have difficulties in dealing with long-term temporal information in some cases due to the gradient exploding or vanishing problem. Thus, a variant of the standard RNN, LSTM network was proposed.

Compared with the standard RNN, LSTM is equipped with an additional memory cell to selectively remember the previous inputs. The scale of historical information that a network can forget or remember is controlled by the memory cell, thereby overcoming the gradient exploding or vanishing problem. Thus, LSTM is more efficient than the standard RNN when dealing with tasks that require very deep structures.

In LSTM, the memory cell c_t and the hidden state h_t can be calculated by the following equations:

$$f_t = \sigma\big(W_f x_t + U_f h_{t-1} + b_f\big), \tag{4.3}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i), \tag{4.4}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o), \tag{4.5}$$

$$c_t = tanh(W_c x_t + U_c h_{t-1} + b_c), \tag{4.6}$$

$$s_t = f_t s_{t-1} + i_t c_t, \tag{4.7}$$

$$h_t = o_t * tanh(s_t), \tag{4.8}$$

where $\sigma(\cdot)$ is an activation function (usually a sigmoid function); i_t, f_t and o_t are the three different gates in the memory cell.

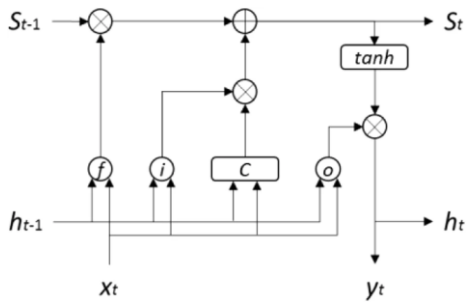


Figure 4.2. Long short-term memory recurrent neural network.

As shown in Figure 4.2, in LSTM, the input gate i_t and the forget gate f_t control whether to remember the current input x_t or forget the previous memory c_{t-1}, and the output gate o_t determines which parts of the history information in the memory cell c_t can be transported to the hidden state h_t. Thus, the collaboration of these three gates allows LSTM to perform or model long-term sequence information.

4.3.2 Caption generation module

The purpose of video captioning is to produce a descriptive sentence $S = \{s_1, s_2, \ldots, s_n\}$, which is able to depict the content of a video $\mathcal{V}$. Conventional encoder-decoder structures usually establish models for the caption generation probability word by word:

$$P(S|V) = \prod_{t=1}^{n} P(s_t|s_{<t}, \mathcal{V}; \theta), \tag{4.9}$$

where n denotes the length of the sentence S; $s_{<t}$ denotes the partial caption that has been generated, *i.e.*, $\{s_1, s_2, \ldots, s_{t-1}\}$; θ denotes the parameters in an encoder-decoder model.

In the encoding stage: To generate reliable video captions, some visual features, which contain the high-level semantic information of a video, need to be captured (*i.e.*, the process of feature extraction). Previous approaches usually leverage CNNs (*e.g.*, AlexNet [21], VGG19 [38], and GoogleNet [28]) for feature extraction since these networks can convert each video frame into a fixed-length video representation that contains high-level semantic information.

Considering that we need a deeper network to extract video representation, in this chapter, the Inception-V4 [39] is introduced as an encoder to extract features from raw videos. Thus, a given video $\mathcal{V}$ can be encoded into a sequence $\{v_1, v_2, \ldots, v_m\}$ as video representation, where m denotes the total frame number of a video.

In the decoding stage: The decoder generates captions word by word according to the video representation. Usually, LSTM, which is capable of modeling long-term temporal dependencies, is utilized as a decoder to convert the video representation into video captions. Moreover, to further exploit the most salient regions in videos, attention mechanisms are often introduced into the decoder, which is used to select the key video frames for captioning.

In this chapter, LSTM is utilized as a decoder to convert video representations into video captions, and a multi-head dot product attention (MHDPA) [40] is employed to help the decoder to exploit the most salient regions in videos.

During the process of video captioning, the word prediction at the time step t is performed by LSTM:

$$P(s_t|s_{<t}, \mathcal{V}, \theta) \propto exp\big(\varphi(s_{t-1}, h_t, e_t; \theta)\big), \tag{4.10}$$

where $\varphi(\cdot)$ denotes an activation function of LSTM; h_t denotes the LSTM hidden state calculated at the time step t; e_t denotes the context vector calculated by MHDPA at the time step t.

Moreover, since we utilize MHDPA to assign attention weight α_j^t to the video representation of each frame $\{v_1, v_2, \ldots, v_m\}$ at the time step t, the t^{th} context vector can be calculated as follows:

$$e_t = \sum_{j=1}^{m} \alpha_j^t v_j, \tag{4.11}$$

where m denotes the frame number of a video.

As demonstrated in [24], the attention mechanisms encourage the decoder to choose a subset of key video frames to produce the most appropriate word at each time step. In other words, all currently generated words are summarized (or memorized) in the $t-1^{\text{th}}$ hidden state h_{t-1}. Then the correlations between the j^{th} feature in the video sequence and all currently produced words can be reflected by the attention weight α_j^t at the time step t.

MHDPA is a self-attention mechanism proposed in [40]. Specifically, it utilizes matrices Q, K, and V to respectively store all queries, keys, and values. All these queries, keys, and values can be built by using a linear projection:

$$Q = MW_q, \tag{4.12}$$

$$K = MW_k, \tag{4.13}$$

$$V = MW_v, \tag{4.14}$$

where W_* denotes weight matrices; M denotes a randomly initialized matrix of memories.

The attention is obtained by calculating a set of queries simultaneously. In other words, the dot products of a query (*i.e.*, dot-product attention) can be computed by all keys K, the dimensionality of the key vectors d_k, and a $softmax$ function [40]. Mathematically,

$$A(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{4.15}$$

where the $softmax$ function is utilized to get the weights on the values. The dot-

product attention can be presented as:

$$A_\theta(M) = softmax\left(\frac{MW_q(MW_k)^T}{\sqrt{d_k}}\right)MW_v, \tag{4.16}$$

where $\theta = (W_q, W_k, W_v)$. The output of $A_\theta(M)$ is a matrix that has the same dimensionality as M, represented as M'. M' is an update of M, in which each element m'_e in M' consists of the information from the matrix of memories M. Therefore, every memory is updated based on the information from other memories at each step of the attention, and the information can be shuttled via the parameters W_q, W_k and W_v from memory to memory.

In this chapter, the proposed caption generation module is jointly trained by minimizing the negative log-likelihood to generate accurate natural language sentences for the given videos. Mathematically,

$$\min_\theta \sum_{t=1}^{N}\{-logP(S^t|\mathcal{V}^t;\theta)\}. \tag{4.17}$$

4.3.3 Video reconstruction module

As shown in Figure 4.1, the proposed video reconstruction module is used to reproduce videos. In other words, it is to generate vectors that can represent the content of video frames according to the hidden state sequence of the decoder. However, it is difficult to directly reproduce video frames using the hidden states in the caption generation module due to the high dimension and diversity of raw video frames. Thus, in this section, the proposed video reconstruction module takes the hidden states sequence of the decoder $H = \{h_1, h_2, \dots, h_n\}$ as input to reproduce the video representations created by the encoder.

The benefits of building such a module is two-fold: (i) with such a video reconstruction process, more useful information can be extracted from raw video sequences by the decoder; (ii) the proposed video reconstruction module is able to be trained in an end-to-end fashion. Thus, the relationships between the raw videos and the generated video captions are able to be further enhanced, so that to improve the accuracy of video captioning.

The proposed video reconstruction module is composed of LSTM and MHDPA.

Thus, for each frame, the video representation can be reproduced by the key hidden states of the decoder which is chosen by MHDPA:

$$\mu_t = \sum_{j=1}^{n} \beta_j^t h_j, \tag{4.18}$$

where β_j^t denotes the attention weight calculated by MHDPA for the j^{th} hidden state at time step t. Thus, the correlations between the j^{th} hidden state in the generated captions and all currently reconstructed video representations $\{z_1, z_2, \dots, z_{t-1}\}$ can be measured by β_j^t. This helps the proposed video reconstruction module to selectively process the hidden states according to the attention weight β_j^t and dynamically generate contextual information μ_t at each time step. Moreover, both the generated context μ_t and the hidden state h_t are used as input. Therefore, the proposed video reconstruction module is able to further employ the word composition and the temporal dynamics of the whole video caption to close the relationships between the raw videos and the generated video captions.

4.3.4 Loss function

In this chapter, since the video representation is produced frame by frame, we define the reconstruction loss function as:

$$L_{rec}^{l} = \frac{1}{m} \sum_{j=1}^{m} \psi\left(z_j, v_j\right), \tag{4.19}$$

where z_j denotes the hidden states of the proposed video reconstruction module; v_j denotes the video representation; $\psi(\cdot)$ denotes the Euclidean distance measure function.

As shown in Equation (4.20), the proposed ADL approach is trained by minimizing the whole loss function. The whole loss function contains two phases: one is a video-to-sentence phase that is calculated by the forward likelihood; the other is a sentence-to-video phase that is calculated by the backward loss function. Thus, the loss function of the proposed approach can be defined as:

$$L(\theta, \theta_{rec}) = \sum_{j=1}^{N} \left(-logP\left(S^j | \mathcal{V}^j; \theta\right) + \lambda L_{rec}\left(\mathcal{V}^j, Z^j; \theta_{rec}\right) \right), \quad (4.20)$$

where the generation loss $-logP\left(S^i | \mathcal{V}^i; \theta\right)$ can be calculated by Equation (4.17); the reconstruction loss $L_{rec}\left(\mathcal{V}^i, Z^i; \theta_{rec}\right)$ can be calculated by Equation (4.19); the hyper-parameter λ is introduced to find a compromise/balance between the proposed caption generation module and the proposed video reconstruction module. The larger the difference between the generated results and the ground truth, the greater the gradient of the loss function, and the faster the convergence rate.

As shown in Algorithm 4.1, the training of the proposed ADL approach can be separated into two phases:

- In the first phase, we train the proposed caption generation module based on the forward likelihood, which terminates the training process according to the early stopping strategy.

- In the second phase, we utilized the whole loss function to jointly train the video reconstruction module and finetune the caption generation module. Both the hidden state sequence and the video frame feature sequence are used to calculate the video reconstruction loss function.

Algorithm 4.1: ADL training algorithm

Input: training pairs <*video*, *ground-truth caption*>
randomly initialize parameters;
extract features from videos using the Inception-V4 network;
for *each epoch* **do**
    generate captions using the proposed caption generation module;
    reconstruct videos using the proposed video reconstruction module;
    calculate the loss function;
end

4.4 Experiments

We evaluate the proposed attention based dual learning (ADL) video captioning approach on two benchmark datasets: Microsoft Research video to text (MSR-VTT) [38] dataset and Microsoft Research Video Description Corpus (MSVD) [41]. To

demonstrate the effectiveness of ADL, we utilize the popular evaluation metrics including METEOR [42], BLEU-4 [43], ROUGE-L [44], and CIDEr [45] with the codes released on the Microsoft COCO evaluation server [46].

4.4.1 Datasets and experimental setting

The details of the two benchmark datasets are shown below:

MSVD: MSVD [41] consists of 1970 YouTube video clips, each of which describes one single activity, and its length is between 10 and 25 seconds. Each video clip was annotated with approximately 40 English captions. Similar to [1], 1200 video clips are used as the training set, 100 video clips are used as the validation set, and 670 video clips are used as the test set in this chapter.

MSR-VTT: MSR-VTT [38] is one of the largest datasets for video captioning so far. In this chapter, the initial version of MSR-VTT (*i.e.*, MSR-VTT-10K) is utilized for experiments. MSR-VTT-10K consists of 10K video clips from 20 categories. Approximately 20 sentences are used to annotate a video clip. In summary, MSR-VTT-10K consists of a total of 29,316 unique words and 200K clip-sentence pairs. In this chapter, similar to [1], 6513 video clips are used as the training set, 497 video clips are used as the validation set, and 2990 video clips are used as the test set.

For the sentences in datasets, we removed punctuations, separated the sentences with blank spaces, and then transformed all words into lowercase. We truncated the sentences longer than 30. For each word, the word embedding size is set as 468.

For the proposed caption generation module, all frames in a video clip are fed into Inception-V4 pre-trained on the ILSVRC2012-CLS classification dataset [47]. In this way, frame features are reshaped to the standard size 299 × 299, so that the semantic feature of each video frame can be extracted from the last pooling layer with 1536 dimensions.

Inspired by [24], for each video clip, 28 equally-spaced features are selected. When the number of features is less than 28, zero vectors are used for filling. Moreover, we set the input dimension of the decoder to 468, which is equal to the dimension of the word embedding. In addition, there are 512 units contained in the hidden layer.

In the video reconstruction module, the hidden state of the decoder is taken as the

input, the dimension of which is set to 512. To simplify the calculation of the reconstruction loss function, we set the size of the hidden layer to the same size as the video presentation, *i.e.*, 1536 dimensions.

Wang *et al*. [1] have verified that for a dual learning-based approach, the hyper-parameter λ can balance the contributions of the two modules (*i.e.*, the caption generation module and the video reconstruction module). Thus, the selection of the hyperparameter λ is crucial. Although they have shown that adding the reconstruction loss is able to enhance the performance of video captioning, a too-large λ may cause an obvious decrease in the performance of video caption generation. Therefore, in this chapter, we set λ to 0.1 based on experiences.

We utilized AdaDelta [48] to optimize the training process. Furthermore, the training process will be stopped, when the CIDEr value on the validation set stopped growing for the next 20 consecutive epochs. Then in the test process, we used a beam search with size 5 to generate the final video captions.

Hardware and Software Environment: The experiments in this chapter are executed on a deep learning workstation with Intel Core i9 CPU, four GTX 1080 Ti GPUs, and 128GB RAM. We implement the proposed approach by Python.

4.4.2 Experimental results

We tested the ADL approach on two benchmark datasets for video captioning. Tables 4.1 and 4.2 show the quantitative experimental results on these two datasets.

On the MSVD dataset, we compared the proposed ADL approach with several classical encoder-decoder approaches and the state-of-the-art approaches, including MP-LSTM [1], GRU-RCN [49], HRNE [50], LSTM-E [47], h-RNN [51], aLSTMs [52], LSTM-LS [53], and RecNet [1], for video captioning. The experimental results are shown in Table 4.1. Compared with the classical encoder-decoder approaches, such as MP-LSTM [1], ADL is able to obtain better evaluation results since it contains an additional video reconstruction module to improve the captioning accuracy.

Furthermore, although the training time/convergence rate of the proposed ADL approach is similar to RecNet [1], the performance of ADL is better than RecNet [1], which also contained a video reconstruction module, for video captioning. This is

because the attention mechanism in the proposed approach can capture important information from videos to generate accurate video captions.

Table 4.1. Experimental results of different video captioning approaches in terms of METEOR, BLEU-4, ROUGE-L, and CIDEr scores on MSVD (%).

Approaches	METEOR	BLEU-4	ROUGE-L	CIDEr
MP-LSTM (AlexNet) [1]	29.1	33.3	-	-
GRU-RCN [49]	31.6	43.3	-	68.0
HRNE [50]	33.1	43.8	-	-
LSTM-E [47]	31.0	45.3	-	-
h-RNN [51]	32.6	49.9	-	65.8
aLSTMs [52]	33.3	50.8	-	74.8
LSTM-LS (VGG19) [53]	31.2	46.5	-	-
$\text{RecNet}_{\text{local}}$ (S2VT) [1]	32.7	43.7	68.6	69.8
$\text{RecNet}_{\text{local}}$ (SA-LSTM) [1]	34.1	52.3	69.8	80.3
ADL	**35.7**	**54.1**	**70.4**	**81.6**

Table 4.2. Experimental results of different video captioning approaches in terms of METEOR, BLEU-4, ROUGE-L, and CIDEr scores on MSR-VTT (%).

Approaches	**METEOR**	**BLEU-4**	**ROUGE-L**	**CIDEr**
MP-LSTM (AlexNet) [1]	23.4	32.3	-	-
MP-LSTM (GoogleNet) [1]	24.6	34.6	-	-
MP-LSMT (VGG19) [1]	24.7	34.8	-	-
SA-LSTM (AlexNet) [1]	23.8	34.8	-	-
SA-LSTM (GoogleNet) [1]	25.2	35.2	-	-
SA-LSTM (VGG19) [1]	25.4	35.6	-	-
SA-LSTM (Inception-V4) [1]	25.5	36.3	58.3	39.9
$\text{RecNet}_{\text{global}}$ (SA-LSTM) [1]	26.2	38.3	59.1	41.7
$\text{RecNet}_{\text{local}}$ (SA-LSTM) [1]	26.6	39.1	59.3	42.7
ADL	**26.6**	**40.2**	**60.2**	**44.0**

On the MSR-VTT dataset, we compared the proposed ADL approach with several classical encoder-decoder approaches and the state-of-the-art approaches, including MP-LSTM [1], SA-LSTM [1], and RecNet [1], for video captioning. Table 4.2 illustrated the quantitative experimental results of these approaches on the MSR-VTT dataset. When using the same encoder (such as AlexNet), the performance of SA-LSTM is better than MP-LSTM. This is because MP-LSTM utilized mean-pooling for frame feature aggregation/fusion, while SA-LSTM was based on an

attention mechanism for feature fusion.

Furthermore, compared to other SA-LSTMs that utilized AlexNet, GoogleNet, or VGG19 as the encoder, SA-LSTM that utilized Inception-V4 as the encoder produced the best captioning results. This is because Inception-V4 is deeper than other networks, and is good at extracting advanced semantic features from videos. Therefore, our ADL approach with Inception-V4 is superior to other reference approaches.

Compared with the result of other reference approaches, our proposed approach leverages the strength of dual learning in video caption generation to improve all evaluation scores. Therefore, our approach can generate accurate captions from videos. Figure 4.3 shows qualitative examples of video captions generated by our approach. We compared the generated captions with the ground truths (GT).

GT: A man is playing an electric guitar.
Ours: A man is playing a guitar.

GT: A woman is frying food.
Ours: A person is cooking.

GT: A woman is dancing while singing.
Ours: A woman is dancing.

Figure 4.3. Qualitative examples of video captions generated by our approach. We compared the generated captions with the ground truths (GT).

4.5 Summary

Video captioning aims to produce natural language sentences from videos, which has

been widely used to solve real-world problems, such as explaining a movie plot to the blind.

In this chapter, we proposed a novel attention based dual learning approach (ADL) for video captioning. The proposed video captioning approach consists of two modules: a caption generation module that can generate video captions from raw video frames and a video reconstruction module that can reproduce the raw video frames based on the generated video captions. A multi-head attention mechanism is used in the two modules to capture the most effective information from raw videos and generated captions, and a dual learning mechanism is used to fine-tune the two modules. Therefore, the proposed approach is able to minimize the semantic gap between raw videos and generated captions by decreasing the differences between the reproduced and raw video features.

We test the proposed approach on two benchmark datasets. Experimental results demonstrate the superiority of our proposed approach over the state-of-the-art approaches for video captioning. Our research also verifies the effectiveness of multi-head attention based dual learning for generating high-quality video captions.

The proposed approach also can be improved. For example, we simply use the multi-head attention than developing a new attention mechanism for capturing information from videos and captions. For future work, we intend to develop more appropriate attention mechanisms and better video captioning approaches for video caption generation. We will also explore the use of other information such as audio and semantic information for video caption generation. In addition, we intend to apply video captioning to a wider application field for solving real-world problems.

References

1. B. Wang, L. Ma, W. Zhang, and W. Liu. Reconstruction network for video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7622-7631. 2018.
2. J. Wang, W. Wang, Y. Huang, L. Wang, and Tieniu Tan. M3: Multimodal memory modeling for video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7512-7520. 2018.
3. C. Yan, Y. Tu, X. Wang, Y. Zhang, X. Hao, Y. Zhang, and Q. Dai. STAT: Spatial-temporal attention mechanism for video captioning. *IEEE Transactions on Multimedia*, 22(1): 229-241. 2019.

4. J. Li, Y. Wong, Q. Zhao, and M. S. Kankanhalli. Video storytelling: Textual summaries for events. *IEEE Transactions on Multimedia*, 22(2): 554-565. 2019.

5. F. Hou, R. Wang, J. He, and Y. Zhou. Improving entity linking even further through semantic reinforced entity embeddings. In *the Annual Conference of the Association for Computational Linguistics*, pp. 1-8. 2020.

6. Z. Liu, Z. Li, M. Zong, W. Ji, R. Wang, and Y. Tian. Spatiotemporal saliency based multi-stream networks for action recognition. In *Asian Conference on Pattern Recognition*, pp. 74-84. 2019.

7. H. Zheng, R. Wang, W. Ji, M. Zong, W. K. Wong, Z. Lai, and H. Lv. Discriminative deep multi-task learning for facial expression recognition. *Information Sciences*, 533: 60-71. 2020.

8. S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, S. Venugopalan, R. Mooney, T. Darrell, and K. Saenko. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2712-2719. 2013.

9. F. Xiao, X. Gong, Y. Zhang, Y. Shen, J. Li, and X. Gao. DAA: Dual LSTMs with adaptive attention for image captioning. *Neurocomputing*, 364: 322-329. 2019.

10. H. Wang, H. Wang, and K. Xu. Evolutionary recurrent neural network for image captioning. *Neurocomputing*, 401: 249-256. 2020.

11. R. Li, H. Liang, Y. Shi, F. Feng, and X. Wang. Dual-CNN: A convolutional language decoder for paragraph image captioning. *Neurocomputing*, 396: 92-101. 2020.

12. A. Kojima, T. Tamura, and K. Fukunaga. Natural language description of human activities from video images based on concept hierarchy of actions. *International Journal of Computer Vision*, 50(2): 171-184. 2002.

13. M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele. Translating video content to natural language descriptions. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 433-440. 2013.

14. R. Xu, C. Xiong, W. Chen, and J. J. Corso. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *the 29th AAAI Conference on Artificial Intelligence*. pp. 1-7. 2015.

15. J. Ma, R. Wang, W. Ji, H. Zheng, E. Zhu, and J. Yin. Relational recurrent neural networks for polyphonic sound event detection. *Multimedia Tools and Applications*, 78(20): 29509-29527. 2019.

16. Y. Wu, X. Ji, W. Ji, Y. Tian, and H. Zhou. CASR: a context-aware residual network for single-image super-resolution. *Neural Computing and Applications*, DOI: https://doi.org/10.1007/s00521-019-04609-8. 2019.

17. M. Zong, R. Wang, Z. Chen, M. Wang, X. Wang, and J. Potgieter. Multi-cue based 3D

residual network for action recognition. *Neural Computing and Applications*, pp. 1-12. 2020.

18. T. Jin, Y. Li, and Z. Zhang. Recurrent convolutional video captioning with global and local attention. *Neurocomputing*, 370: 118-127. 2019.

19. Z. Wu, T. Yao, Y. Fu, and Y. Jiang. Deep learning for video classification and captioning. In *Frontiers of Multimedia Research*, pp. 3-29. 2017.

20. W. Zhao, W. Xu, M. Yang, J. Ye, Z. Zhao, Y. Feng, and Y. Qiao. Dual learning for cross-domain image captioning. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 29-38. 2017.

21. S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko. Translating videos to natural language using deep recurrent neural networks. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1494-1504. 2015.

22. S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4534-4542. 2015.

23. C. Zhang, and Y. Tian. Automatic video description generation via LSTM with joint two-stream encoding. In *the 23rd International Conference on Pattern Recognition*, pp. 2924-2929. 2016.

24. L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. Describing videos by exploiting temporal structure. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4507-4515. 2015.

25. D. He, Y. Xia, T. Qin, L. Wang, N. Yu, T. Liu, and W. Ma. Dual learning for machine translation. In *Advances in Neural Information Processing Systems*, pp. 820-828. 2016.

26. Y. Wang, Y. Xia, L. Zhao, J. Bian, T. Qin, G. Liu, and T. Liu. Dual transfer learning for neural machine translation with marginal distribution regularization. In *the 32nd AAAI Conference on Artificial Intelligence*, pp. 1-7. 2018.

27. G. Lample, A. Conneau, L. Denoyer, and M. A. Ranzato. Unsupervised machine translation using monolingual corpora only. In *International Conference on Learning Representations*, pp. 1-10. 2018.

28. M. Artetxe, G. Labaka, E. Agirre, and K. Cho. Unsupervised neural machine translation. In *International Conference on Learning Representations*, pp. 1-10. 2018.

29. Z. Yi, H. Zhang, P. Tan, and M. Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2849-2857. 2017.

30. T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim. Learning to discover cross-domain relations with generative adversarial networks. In *Proceedings of the 34th International*

Conference on Machine Learning, 70: 1857-1865. 2017.

31. J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2223-2232. 2017.

32. Y. Xia, J. Bian, T. Qin, N. Yu, and T. Liu. Dual Inference for Machine Learning. In *International Joint Conferences on Artificial Intelligence*, pp. 3112-3118. 2017.

33. P. Luo, G. Wang, L. Lin, and X. Wang. Deep dual learning for semantic image segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2718-2726. 2017.

34. Y. Wang, Y. Xia, T. He, F. Tian, T. Qin, C. Zhai, and T. Liu. Multi-agent dual learning. In *Proceedings of the International Conference on Learning Representations*, pp. 1-15. 2019.

35. Z. Zhao, Y. Xia, T. Qin, and T. Liu. Dual learning: Theoretical study and algorithmic extensions. In *Proceedings of the International Conference on Learning Representations*, pp. 1-16. 2019.

36. Y. Xia, T. Qin, W. Chen, J. Bian, N. Yu, and T. Liu. Dual supervised learning. In *Proceedings of the 34th International Conference on Machine Learning*, 70: 3789-3798. 2017.

37. Y. Xia, X. Tan, F. Tian, T. Qin, N. Yu, and T. Liu. Model-level dual learning. In *International Conference on Machine Learning*, pp. 5383-5392. 2018.

38. J. Xu, T. Mei, T. Yao, and Y. Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5288-5296. 2016.

39. C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *the 31st AAAI Conference on Artificial Intelligence*, pp. 1-7. 2017.

40. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pp. 5998-6008. 2017.

41. D. L. Chen, and W. B. Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 1: 190-200. 2011.

42. S. Banerjee, and A. Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65-72. 2005.

43. K. Papineni, S. Roukos, T. Ward, and W. Zhu. BLEU: a method for automatic evaluation

of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 311-318. 2002.

44. C. Lin. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74-81. 2004.

45. R. Vedantam, C. L. Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4566-4575. 2015.

46. X. Chen, H. Fang, T. Lin, R. Vedantam, S. Gupta, P. Dollár, and C. L. Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*. 2015.

47. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang *et al*. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3): 211-252. 2015.

48. M. D. Zeiler. ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*. 2012.

49. N. Ballas, L. Yao, C. Pal, and A. Courville. Delving deeper into convolutional networks for learning video representations. In *International Conference on Learning Representations*, pp. 1-10. 2016.

50. P. Pan, Z. Xu, Y. Yang, F. Wu, and Y. Zhuang. Hierarchical recurrent neural encoder for video representation with application to captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1029-1038. 2016.

51. H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu. Video paragraph captioning using hierarchical recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4584-4593. 2016.

52. L. Gao, Z. Guo, H. Zhang, X. Xu, and H. T. Shen. Video captioning with attention-based LSTM and semantic consistency. *IEEE Transactions on Multimedia*, 19(9): 2045-2055. 2017.

53. Y. Liu, X. Li, and Z. Shi. Video captioning with listwise supervision. In *the 31st AAAI Conference on Artificial Intelligence*, pp. 1-7. 2017.

Chapter 5 A bidirectional relational recurrent neural network

Dense video captioning is an emerging task in the field of digital video processing, which aims to locate and describe all events in a video to generate a set of informative sentences describing the video content. However, previous approaches generated dense video captions mainly based on the visual information in a video, which could not locate and describe long-lasting and/or overlapping events. This chapter proposes a novel Bidirectional relational recurrent neural network (Bidirectional RRNN) for dense video captioning with an encoder-decoder structure, which fully utilizes the local and global context information and visual information in the video. Specifically, a 3D convolutional neural network is used to extract visual information from the video. Then a bidirectional RRNN encoder is developed to obtain the local and global context information of a target event, which has a relational memory core for collecting and relational reasoning of temporal context information. Therefore, the proposed approach can capture all events in the video, reason the relationships between these events, and then generate a set of informative sentences to describe the video content. Experimental results demonstrate that the proposed approach outperforms the state-of-the-art approaches on the ActivityNet Captions dataset.

This chapter is organized as follows. Section 5.1 introduces the background of dense video captioning and the motivation of this research. Section 5.2 reviews the most relevant works of this research. Section 5.3 presents the proposed approach in detail. Section 5.4 presents the experiments and discusses the experimental results. At the end of this chapter, the conclusion of this research is presented in Section 5.5.

5.1 Introduction

Dense video captioning is an emerging task in the field of digital video processing, which aims to locate and describe all events in a video to generate a set of informative sentences describing the video content [1]. It bridges computer vision and natural languages, and has a wide range of applications, such as describing videos (*e.g.*, movies) to the blind, and improving the index and search quality of online videos.

Previous research on video captioning focused on producing a descriptive sentence for a short video (*e.g.*, a 10-second video in the MSVD dataset [2]). However, most

real-world videos are long videos, which contain multiple events entangled along the timeline, and last for much longer than a few seconds. Compared with short videos, long videos usually involve more objects, scenes, events, temporal relations, and so on. Therefore, various dense video captioning approaches were developed to capture and describe video events.

Events in a video usually have different durations and span different time scales on the timeline. Previous research on dense video captioning simply utilized sliding windows to capture video events, that is, sliding a window on a video sequence to classify the content in each window as a background or event. However, the video information that can be captured by the sliding windows is limited [1]. Current dense video captioning approaches mainly utilize the mean-pooling [3] or recurrent neural networks (RNNs) [4] to encode the entire video sequence to capture video events. However, these approaches are only effective for processing short videos. Encoding a video sequence spanning a few minutes may lead to the vanishing gradient problem, making it impossible to describe all events in the video accurately.

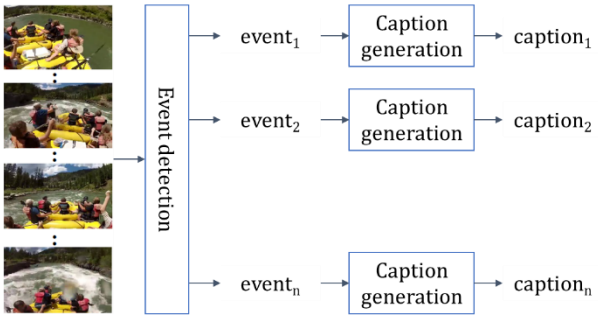
Since the events in a video are usually interconnected, Yu *et al*. [5] developed a paragraph captioning approach based on hierarchical recurrent neural networks (h-RNNs), which described video contents by generating a paragraph containing multiple sentences. Both visual information and historical paragraph information were used as the input of their proposed sentence generator to capture context information. However, this approach only explored the use of past context information on paragraph generation, and the generated paragraphs can only describe the video content sequentially, *i.e*., describing the video content in a chronological order of events that occurred. If the events in a video overlap each other on the timeline, the performance of this approach will be unsatisfactory since it cannot accurately separate the overlapping events in videos [1].

More recently, Wang *et al*. [6] proposed a bidirectional attentive fusion approach for dense video captioning, which can generate event proposals for videos, and fuse the hidden states of event proposals with video features to generate dense captions. Their approach can separate the overlapping events in videos, but does not work well when dealing with long-lasting events (*e.g*., an event that lasts almost the same duration as the entire video).

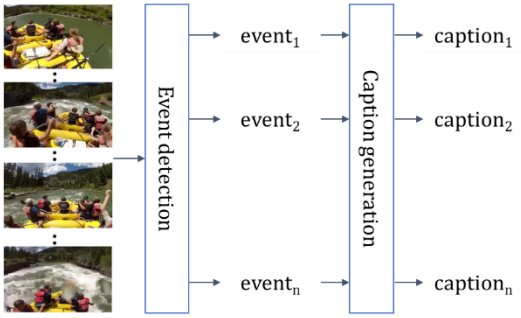
In this chapter, a novel Bidirectional Relational Recurrent Neural Network (Bidirectional RRNN) is proposed for dense video captioning with an encoder-

decoder structure, which can generate a set of informative sentences to describe all events in a video, including long-lasting and/or overlapping events.

Figure 5.1 compares the proposed approach with previous dense video captioning approaches. A common framework of the previous approaches is shown in Figure 5.1(a), which generates video captions for each event detected/captured from a video, respectively. Different from these approaches, the proposed Bidirectional RRNN approach (shown in Figure 5.1(b)) fully utilizes the local and global context information and visual information in the video to generate realistic dense captions.



(a) Previous dense video captioning approaches.



(b) The proposed dense video captioning approach.

Figure 5.1. Comparison between (a) the common framework of previous dense video captioning approaches and (b) the proposed dense video captioning approach.

In Bidirectional RRNN, a 3D convolutional neural network is used to extract visual information from the video. Then a bidirectional RRNN encoder is developed to obtain the local and global context information of a target event, which utilizes a relational memory core [23] for collecting and relational reasoning of temporal context information. Therefore, the Bidirectional RRNN approach is capable of capturing all events in the video, including long-lasting and/or overlapping events, reasoning the relationships between these events, and then generating a set of informative sentences to describe the video content. We test the proposed approach

on the ActivityNet Captions dataset [9]. Experimental results demonstrate that the proposed Bidirectional RRNN approach is superior to the state-of-the-art approaches.

5.2 Related work

In this section, the related work for video captioning and dense video captioning is presented.

5.2.1 Video captioning

Video captioning is to produce a descriptive sentence from a video [7]. Existing approaches for video captioning are mainly based on two types of models: (i) template-based language models, and (ii) sequence learning-based models.

The template-based language models can align the video content with each fragment of the predefined template to generate video captions [8-11], which can generate captions by using a series of predefined language templates that can be separated into several fragments (*e.g.*, object, verb, and subject) according to specific grammatical rules [12]. In the template-based language models, object detection techniques were used to map features extracted from a video to words. Then the mapped words were placed into different fragments of a predefined template according to the grammatical rules to generate captions describing the video content.

The following are some typical examples of the template-based language models. Guadarrama *et al*. [8] developed an activity recognition approach named zero-shot recognition to describe video activities, and introduced the concept of semantic hierarchy to learn the semantic relationships between different sentence fragments. Similarly, Kojima *et al*. [9] proposed the concept hierarchy of human actions, which was used to describe human activities in videos. Different from the above works, Rohrbach *et al*. [10] utilized a conditional random field (CRF) to simulate the connotations between activities and objects in videos, and developed a video captioning approach based on visual features and semantic features. In addition, Xu *et al*. [11] proposed a video captioning approach consisted of three modules (*i.e.*, a joint embedding module, a deep video module, and a semantic language module) to generate natural language sentences from videos.

However, since the sentences generated by the template-based language models are

highly dependent on the objects/words detected from a video, only part of video content can be textualized, *i.e.*, only limited words can be mapped. Further, since the generated sentences are highly dependent on the predefined templates, the generated video captions were based on simple and uniform syntactic structures, and even be robotic in some cases [12].

In order to solve the above problems, sequence learning-based models, also known as the encoder-decoder model, was introduced for video captioning, which can generate sentences with flexibly syntactical structures [12]. The sequence learning-based model contains two stages: the encoding stage and the decoding stage. Usually, convolutional neural networks (CNNs) are used as the encoder to extract features from videos, and recurrent neural networks (RNNs) are used as the decoder to generate captions word-by-word based on the extracted features.

The following are some typical examples of sequence learning-based models. Venugopalan *et al*. [3] developed a video captioning approach, which extracted visual features from videos using CNNs and fed the averaged CNN features into a Long Short Term Memory (LSTM) network to generate video captions. Venugopalan *et al*. [4] also developed a well-known approach named Sequence to Sequence Video to Text (S2VT), which captured the temporal dynamics of video sequences during video captioning. The proposed S2VT approach extracted motion and appearance features using CNNs from optical flow and still image frames, respectively, and generated video captions by using LSTMs. After that, Zhang and Tian [13] developed a two-stream neural network to explore the effect of spatial and temporal information on generating video captions.

Recently, attention mechanisms have also been introduced to video captioning approaches, which have been proven to be an effective way to enhance the performance of video captioning approaches. The attention-based video captioning approaches can focus on the important information in a video while ignoring other irrelevant information, improving the accuracy of the generated captions. For example, Yao *et al*. [14] developed an attention-based video captioning approach, which assigned weights to the visual features using an attention mechanism. Then the video captions were generated based on the weighted features. The performance of their approach is better than the performance of other video captioning approaches without attention mechanisms.

To summarize, the encoder-decoder models have become the mainstream for video

captioning since they allow the training process works in an end-to-end manner. The existing video captioning approaches have made significant achievements in dealing with short videos. However, most real-world videos are long videos, which consist of multiple events untangled along the timeline. Therefore, a variety of dense video captioning approaches was developed to deal with these long videos.

5.2.2 Dense video captioning

The key issue in dense video captioning is to locate and describe all events in videos.

An early attempt on dense video captioning was h-RNN [5], which used hierarchical recurrent neural networks to generate dense captions for videos. Both video features and historical paragraph states (generated by embedding previously generated sentences) were used as the inputs to the sentence generator. A temporal attention mechanism was also introduced into the sentence generator to improve the performance of the sentence generator. However, this approach cannot align the generated sentences with the video contents [15].

To solve the above problem, Shen *et al*. [15] developed a weakly supervised dense video captioning approach that could generate dense captions for videos with video-level annotations for training. Specifically, the proposed approach consisted of three components: (i) a visual sub-model, which built a weak mapping between the words in annotations and the regions in video frames using multi-instance multi-label learning; (ii) a region-sequence sub-model, which generated informative region-sequences by matching and sequentially connecting the regions between different video frames based on the output of the visual sub-model; (iii) a language sub-model, which generated dense video captions based on the region-sequences. This work was also the first time to introduce multi-instance learning into dense video captioning. However, this approach did not perform well in processing long-lasting and/or overlapping events [ref].

Krishna *et al*. [1] created the ActivityNet Captions dataset, that contains 20k long videos from ActivityNet [16]. Each video in the ActivityNet Captions dataset was annotated with a series of temporally localized descriptive sentences. They also defined a concept named video proposal (*i.e*., temporally localized video segments) and proposed a baseline for dense event captioning, which has been wildly used to evaluate the quality of a generated dense caption. In their approach, the video proposals were generated by using an action proposal approach named deep action

proposal (DAP) [17] with an attention mechanism. The generated proposals were then fed into an LSTM caption generator to generate video captions.

Similarly, Li *et al*. [18] also developed a proposal-based dense video captioning approach, which consisted of two modules: a temporal event proposal (TEP) module and a sentence generation (SG) module. In the TEP module, a convolutional structure was used to perform event/background classification, temporal boundaries refinement, and descriptive regression for each proposal. Then, the refined proposals and their visual information were fed to the SG module, achieved by LSTM networks, for dense caption generation. Further, reinforcement learning was used to train the SG module to maximize METEOR scores.

Recently, Wang *et al*. [6] proposed a proposal-based dense video captioning approach, in which event proposals were generated by using a single-stream temporal action proposal (SST [19]). To utilize context information for event localization, they developed a context gating mechanism, which is similar to the gating mechanism in LSTM, to measure the contribution of context information. Then the generated event proposals were fused with visual features for dense video caption generation. Their approach can separate the overlapping events in videos and generate corresponding descriptions, but it cannot work well when dealing with long-lasting events.

To summarize, current approaches for dense video captioning [1,6,17-20] mainly rely on video events, but ignoring the context information in videos. These approaches generate video proposals to capture the events in a video first, and then describe each event to form dense video captions. Such approaches can perform well in capturing simple events. However, in some cases, such as separating overlapping and/or long-lasting events from videos, these approaches cannot achieve comparable performance to humans. To solve this problem, this chapter aims to develop a novel dense video captioning approach that can fully use both the context information and visual information to generate a set of informative sentences to describe video events, including long-lasting and/or overlapping events.

5.3 The proposed approach

This section details the proposed Bidirectional RRNN approach, which consists of two modules: a proposal generation module (Section 5.3.1) and a caption generation module (Section 5.3.2), as illustrated in Figure 5.2. The two modules will be coupled

together and trained in an end-to-end manner. Section 5.3.3 presents the loss functions for training the whole approach.

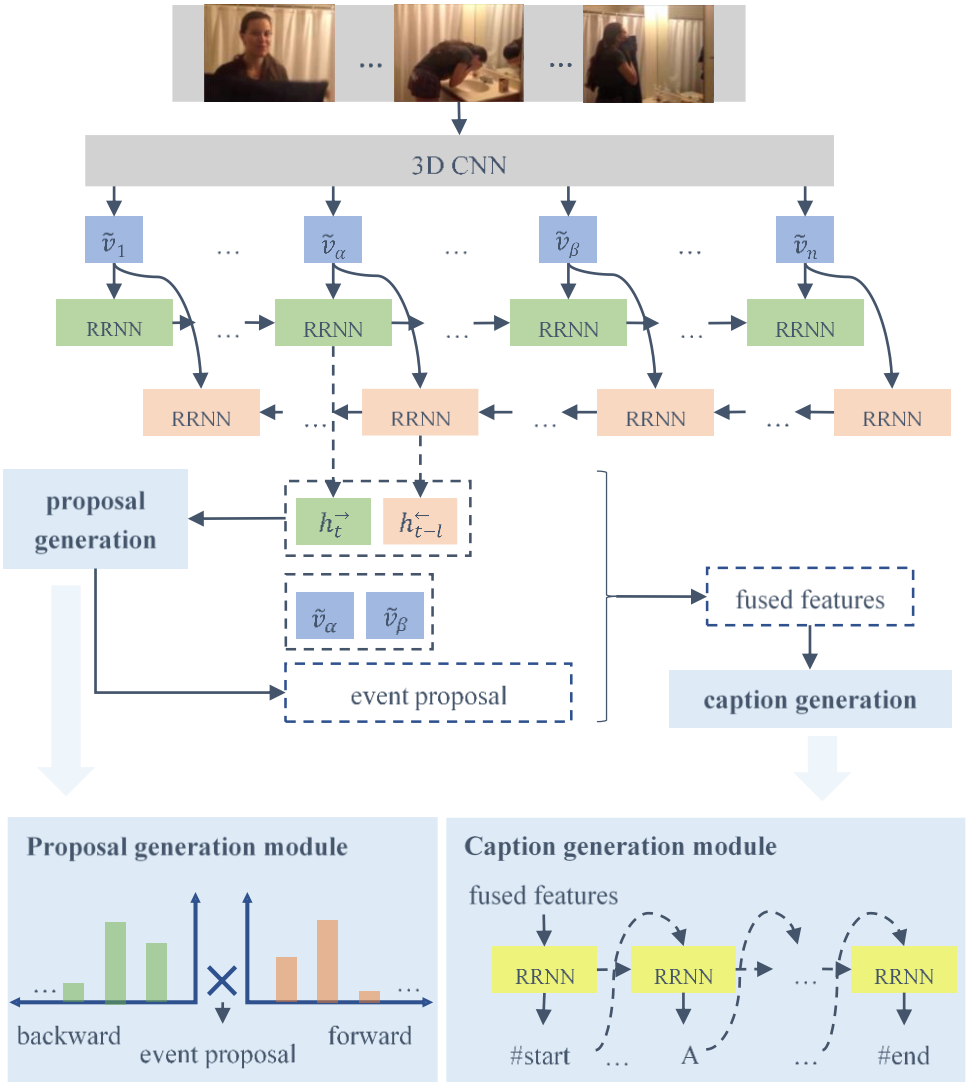


Figure 5.2. Illustration of the proposed dense video captioning approach.

5.3.1 Proposal generation module

The proposal generation module aims to produce a set of temporal regions containing events from videos, which involves three phases: a forward phase, a backward phase, and a fusion phase. In the forward and backward phases, a forward event proposal and a backward event proposal are generated based on RRNNs for the same video,

respectively. Then, in the fusion phase, the outputs of the above two phases will be fused by a context fusion strategy to generate a final event proposal.

Feature extraction. Assume that a video sequence V contains L video frames, *i.e.*, $V = \{v_1, v_2, \ldots, v_L\}$. Following the parameters in [6], we use a 3D convolutional neural network (3D CNN [21]) to encode each video frame for C3D feature extraction. The 3D CNN was pre-trained on the Sports-1M video dataset [22].

The dimensionality of each extracted feature is reduced by using principal component analysis (PCA), from 4096 to 500. The temporal resolution of the extracted C3D features is $\delta = 16$ video frames, which discretizes the input video sequences into $T = L/\delta$ time steps. Hence, the pre-processed video sequence for the given video is presented as $\tilde{V} = \{\tilde{v}_1, \tilde{v}_2, \ldots, \tilde{v}_T\}$. This sequence is then fed into the proposed bidirectional RRNN encoder sequentially.

RRNN. RRNN [23] is a variant of RNN, which is able to achieve complex relational reasoning with 'remembered' information. This is because the memory cell structure of a standard LSTM is replaced by a relational memory core (RMC) in RRNN [24].

Similar to LSTM, an RRNN unit contains a matrix of memories M and three gates (*i.e.*, an input gate i_t, a forget gate f_t, and an output gate o_t). The memory matrix M is randomly initialized and can be considered as a matrix that replaces the cell states C of a standard LSTM. In other words, at time step t, operations on each $c_{e,t}$ are replaced by the operations on each $m_{e,t}$, which is the e^{th} row of M.

Since a multi-head attention mechanism is used in the relational memory core of RRNN to weight the previous memory matrix according to the input vectors. Thus, M' stores the memories processed by the multi-head attention, *i.e.*, an update of M, and $m'_{e,t}$ presents the updated memories stored in the e^{th} row of M' at time step t.

Using RRNN, the hidden activation sequence $\{h_1, h_2, \ldots, h_T\}$ of an input sequence $\{\tilde{v}_1, \tilde{v}_2, \ldots, \tilde{v}_T\}$ can be calculated as follows:

$$s_{e,t} = \left(h_{e,t-1}, m_{e,t-1}\right), \tag{5.1}$$

$$f_{e,t} = \sigma\left(W_f \tilde{v}_t + U_f h_{e,t-1} + b_f\right), \tag{5.2}$$

$$i_{e,t} = \sigma\left(W_i \tilde{v}_t + U_i h_{e,t-1} + b_i\right), \tag{5.3}$$

$$o_{e,t} = \sigma\left(W_o \tilde{v}_t + U_o h_{e,t-1} + b_o\right), \tag{5.4}$$

where $s_{e,t}$ denotes the e^{th} hidden state s_t at time step t; parameter σ is a sigmoid function that maps the input values into the interval (0, 1) to calculate the proportion of information that can get through the three gates; matrices W_* and U_* present the weights connecting any two layers; b_* is a bias term.

Thus, at time step t, the memory cell $m_{e,t}$ and the hidden state $h_{e,t}$ can be calculated by using:

$$m_{e,t} = f_{e,t} m_{e,t-1} + i_{e,t} g_\varphi\left(m'_{e,t}\right), \tag{5.5}$$

$$h_{e,t} = o_{e,t} \cdot tanh\left(m_{e,t}\right), \tag{5.6}$$

where g_φ presents a post-attention processor [23] and *tanh* is an activation function. In this chapter, we set the number of attention head to 2.

Forward. In the forward phase, an RRNN is used as a forward encoder to sequentially encode the video sequence $\tilde{V}$. The proposed encoder is capable of processing video sequences and accumulating visual information across the timeline.

At time step t, the hidden state of the forward encoder is $h_t^{\rightarrow} \in \left\{h_j^{\rightarrow}\right\}_{j=1}^{T}$. Using the hidden states as the input, the K forward confidence scores $C_t^{\rightarrow} = \left\{c_t^{\vec{k}}\right\}_{k=1,\ldots,K}$, which indicate the probabilities of K proposals in the forward phase, is produced using K independent binary classifiers. The forward confidence scores $C_t^{\rightarrow}$ can be calculated by using a fully connected layer:

$$C_t^{\rightarrow} = \sigma(W_c^{\rightarrow} h_t^{\rightarrow} + b_c^{\rightarrow}), \tag{5.7}$$

where σ presents a sigmoid nonlinearity, $W_c^{\rightarrow}$ and $b_c^{\rightarrow}$ are weights and biases that are shared across all time steps.

The K proposals are specified by $S_t^{\rightarrow} = \left\{s_t^{\vec{k}}\right\}_{k=1,\ldots,K}$, where $s_t^{\vec{k}}$ represents a video clip in the given video started at time $t - l^k$ and ended at time step t. Note that, at time step t, the length of the k^{th} predefined proposal anchor is $l^k (k = 1, \ldots, K)$, and the same end time step t is shared by all K proposals in S_t.

Backward. Since the forward phase aims to capture clues for past events, the proposed backward phase aims to capture clues for future events in the video. Making full use of both past event clues and future event clues will lead to more accurate event proposals. Hence, in the backward phase, the video sequence $\tilde{V}$ will be fed into a backward encoder in a reverse order. The backward encoder is also achieved by RRNNs.

Similar to the forward phase, at time step t, the hidden state of the backward encoder is presented as $h_t^{\leftarrow} \in \left\{h_j^{\leftarrow}\right\}_{j=1}^{T}$. The K proposals $S_t^{\leftarrow} = \left\{s_t^{\overleftarrow{k}}\right\}_{k=1,\ldots,K}$ can be produced by K backward confidence scores $C_t^{\leftarrow} = \left\{c_t^{\overleftarrow{k}}\right\}_{k=1,\ldots,K}$.

Fusion. After the above two phases, N proposals can be collected from all time steps in both directions (*i.e.*, forward and backward). To improve the confidence/accuracy of the generated event proposals, we combine the two scores obtained in the two directions for the same proposals to obtain the final confidence scores.

To fuse the proposals from the forward and backward directions, various fusion strategies can be utilized. Since the approach in [6] is a baseline of our approach, this chapter follows the fusing strategy in [6] and uses multiplication to fuse the proposals from both directions by using:

$$C_p = \left\{c_t^{\vec{k}} \times c_t^{\overleftarrow{k}}\right\}_{k=1}^{N}. \tag{5.8}$$

If the final confidence score of a proposal is greater than a threshold τ, the proposal will be selected for further caption generation, otherwise, the proposal will be discarded. Note that since the events in videos may be overlapping and/or long-lasting, a threshold rather than a non-maximum suppression strategy [34] is applied for proposal selection.

5.3.2 Caption generation module

This section will focus on the proposed caption generation module. Following the encoder-decoder architecture, our caption generation module aims to generate multiple sentences from videos to densely describe video contents.

In order to generate video captions based on the generated event proposals, previous work usually fed the hidden states of the proposals into a caption generator. In the proposed approach, the proposal hidden states of the forward and backward phases, as well as the video features will be fused and fed into the proposed caption generation module. Thus, both past and future context information can be integrated for dense caption generation. Formally, the input of the proposed caption generation module is:

$$\mathcal{H}_t(p_n) = \mathcal{F}\left(h_\alpha^{\rightarrow}, h_\beta^{\leftarrow}, \widetilde{V'} = \{\widetilde{v_\alpha}, \dots, \widetilde{v_\beta}\}, \mathcal{H}_{t-1}\right), \tag{5.9}$$

where $\mathcal{F}(\cdot)$ is a mapping that can output a compact vector; α and β denote the start and end time steps of a detected proposal p_n ($n = 1, 2, \dots, N$) (*i.e.*, the n^{th} video event); $h_\alpha^{\rightarrow}$ and $h_\beta^{\leftarrow}$ denote the proposal hidden states of forward and backward phases; $\widetilde{V'} = \{\widetilde{v_\alpha}, \dots, \widetilde{v_\beta}\}$ is the C3D features of the video clip corresponding to proposal p_k; $\mathcal{H}_{t-1}$ denotes the RRNN hidden state at time step t-1.

The most straightforward way to fuse $h_\alpha^{\rightarrow}$, $h_\beta^{\leftarrow}$, and $\widetilde{V'}$ together is concatenation. However, since the dimension of $\widetilde{V'}$ relies on the length of the detected video event, it is not feasible to concatenate them all directly. Other research works tried to concatenate the mean of $\widetilde{V'}$ with proposal hidden states. However, the mean pooling cannot explicitly present the relationship between an event and its context information [20]. Recently, an attention mechanism is introduced to our module to fuse $h_\alpha^{\rightarrow}$, $h_\beta^{\leftarrow}$, and $\widetilde{V'}$ together for caption generation, which is initialized with the random initialization. At time step t, the attention mechanism can be presented as:

$$z_t^j = tanh\left(W_v \tilde{v}_{\alpha+j-1} + W_h\left[h_\alpha^{\rightarrow}, h_\beta^{\leftarrow}\right] + W_{\mathcal{H}}\mathcal{H}_{t-1} + b\right), \tag{5.10}$$

where $[\cdot,\cdot]$ presents the vector concatenation of $h_\alpha^{\rightarrow}$ and $h_\beta^{\leftarrow}$. The weights of $v_{\alpha+j-1}$

can be obtained by a *softmax* normalization:

$$\gamma_t^j = \frac{exp\left(z_t^j\right)}{\sum_{k=1}^{q} exp\left(z_t^k\right)}, \tag{5.11}$$

where $q = \beta - \alpha + 1$ is the length of an event proposal. Then the attended visual feature can be calculated by the following weighted sum:

$$\tilde{v}_t = \sum_{j=1}^{q} \gamma_t^j \cdot v_{\alpha+j-1}, \tag{5.12}$$

Our proposed module can focus on "keyframes" in videos and generate semantic-related words by using the information from context vectors with the introduced attention. Thus, the final input of the proposed caption generation module can be presented as:

$$\mathcal{H}(p_n) = \left[\tilde{v}_t, \overrightarrow{h_\alpha}, \overleftarrow{h_\beta}\right]. \tag{5.13}$$

5.3.3 Loss functions

The proposed approach is a couple of our proposal generation module and the caption generation module. Thus, two types of loss functions (*i.e.*, the proposal generation loss and caption generation loss) are involved for model training.

Proposal generation loss. Following the settings in [19], all ground-truth proposals are first collected and grouped into K=128 clusters (*i.e.*, anchors). We then associate each training sample $\tilde{V} = \{\tilde{v}_1, \tilde{v}_2, \ldots, \tilde{v}_T\}$ with its ground-truth labels $\{y_t\}_{j=1}^{T}$. Each y_t is a K-dimensional vector with binary entries. If the value of the temporal Intersection-over-Union (tIoU) for the k^{th} proposal with ground-truth is exceeded 0.5, $y_t^k (k = 1, \ldots, K)$ will be set to 1, otherwise, it will be set to 0.

According to the parameter setting in [19], to balance negative and positive proposals, we introduced a weighted multi-label cross-entropy as the proposal generation loss $\mathcal{L}_p$. At time step t, for the given video V, the proposal generation loss $\mathcal{L}_p$ is:

$$\mathcal{L}_p(c,t,V,y) = -\sum_{j=1}^{K} w_0^j y_t^j log c_t^j + w_i^j \left(1 - y_t^j\right) log\left(1 - c_t^j\right), \quad (5.14)$$

where w_0^j and w_1^j are determined by the amount of negative and positive proposals, and c_t^k denotes the k^{th} proposal's prediction score at time step t. Both forward and backward losses are calculated in the same way. The results of the forward and backward losses will be added together for jointly training our proposal generation module. Thus, by averaging all time steps, $\mathcal{L}_p$ is then calculated.

Caption generation loss. For the proposed caption generation module, only the proposals that have high tIoU value (>0.8) with ground-truths can be used for training. For a sentence with M words, the caption generation loss $\mathcal{L}_c$ is defined as the sum of the negative log-likelihood of accurate words based on the setting in [25]:

$$\mathcal{L}_c(P) = -\sum_{j=1}^{M} log\left(p\left(w_j\right)\right), \quad (5.15)$$

where w_j is the j^{th} word of a ground-truth sentence. Thus, $\mathcal{L}_c$ can be calculated by averaging all proposals' caption generation loss $\mathcal{L}_c(P)$.

Total loss. In this chapter, the total loss of the proposed approach can be calculated by using both proposal generation loss and caption generation loss:

$$\mathcal{L} = \lambda \mathcal{L}_p + \mathcal{L}_c(P), \quad (5.16)$$

where λ is utilized to balance the contributions between the proposal generation loss and caption generation loss. We simply set it to 0.5.

5.4 Experiments

In this section, we train and test the proposed Bidirectional RRNN approach for dense video captioning. Section 5.4.1 introduces dataset and our experimental settings. Section 5.4.2 presents and discusses our experimental results.

5.4.1 Dataset and experimental setting

To verify the effectiveness of the proposed Bidirectional RRNN approach, we train and test the proposed approach on the ActivityNet Captions dataset [1]. Several popular evaluation metrics including METEOR [26], BLEU-4 [27], ROUGE-L [28], and CIDEr [29] with the codes released on the Microsoft COCO evaluation server [30] are used to evaluate the accuracy of the generated dense captions.

ActivityNet Captions. ActivityNet Captions [1] is a dense video captioning dataset that is created on ActivityNet v1.3 [16]. It contains 20,000 YouTube untrimmed videos from the real world. The average length of these videos is 120 seconds. Most videos contain more than three annotated video events, which are labeled with corresponding start/end times and human-written descriptive sentences. The length of these sentences is 13.5 words on average.

METEOR. METEOR is a popular evaluation metric that has been widely used to measure the similarity between sentences. METEOR has been shown to produce the closest results to human judgments when only a few sentence references are given [28].

Following the setting in [6], when describing the top 1000 proposals for each video, the METEOR scores are averaged at tIoU thresholds of 0.3, 0.5, 0.7, and 0.9 to measure the proposed dense video captioning approach. In addition, the BLEU, CIDEr-D, and Rouge-L scores are also calculated for comparison.

Hardware and Software Environment. The experiments are executed on a deep learning workstation with four GTX 1080 Ti GPUs, 128GB RAM, and Intel Core i9 CPU. The proposed approach is implemented by Python with the DeepMind Sonnet library [http://github.com/deepmind/sonnet], which is a deep learning library built on top of TensorFlow [31].

Reference approaches. We compared the experimental results of the proposed approach with the following reference approaches:

- Bi-SST + H [6]: In this approach, Bi-SST was used to generate proposals, and the hidden states of both directions corresponding to the proposals were concatenated to represent an event for dense caption generation.

- Bi-SST + E [6]: In this approach, Bi-SST was used to generate proposals, and the hidden states corresponding to the proposals were mean pooled to represent an event for dense caption generation.

- H-RNN [5]: This approach generated dense captions from videos by using two RNNs, in which one RNN was utilized to generate individual sentences, while the other one was utilized to sequentially initialize hidden states for generating the next sentence.

- Krishna *et al*. [1]: This approach generated dense video captions by using DAP for proposal generation and an LSTM network for caption generation.

- Wang *et al*. [6]: This approach used a Bi-SST for generating generate event proposals, and the bidirectional hidden states corresponding to the proposals were integrated with video clip features for generating dense captions.

- Li *et al*. [18]: This approach unified the temporal localization and sentence generation of event proposals based on the developed descriptiveness regression for dense video captioning.

- Mun *et al*. [32]: This approach simulated temporal dependency between events in a video explicitly, and used linguistic and visual context from the previous events to perform coherent dense video captions.

- Iashin *et al*. [33]: This approach was based on a transformer architecture that encoded the feature representation of each modality (*e.g*., audio, speech, and visual modalities) for a specific event proposal and produces video captions using the information from these modalities.

5.4.2 Experimental results

In this chapter, the numbers of videos in the training set, validation set, and test set are 10024, 4926, and 5044, respectively. All ground-truth annotations will be retained for competition. We reported METEOR, BLEU, ROUGE-L, and CIDEr-D scores for evaluating the performance of different dense video captioning approaches in Table 5.1.

Table 5.1 reports the results of the proposed approach and reference approaches on the ActivityNet Captions dataset. Several popular evaluation metrics, including METEOR, BLEU, ROUGE-L, and CIDEr-D, are used to evaluate the quality of the generated dense captions. Since dense video captioning is a task that aims to describe all events in a video rather than retell the ground-truth captions, and the METEOR

score is highly consistent with human judgments with a few reference sentences, the METEOR score is more important than the others. Thus, this chapter mainly refers to the METEOR scores for comparison. Detailly, only one reference sentence is involved in this chapter. The experimental results show that our approach with the highest METEOR score outperforms other reference approaches.

Table 5.1. Experimental results of different dense video captioning approaches on the ActivityNet Captions dataset.

Approaches	**METEOR**	**ROUGE-L**	**CIDEr-D**
Bi-SST + H [6]	8.35	17.56	8.17
Bi-SST + E [6]	8.36	17.96	9.13
GT proposals + H-RNN [5]	8.02	-	20.18
learnt proposals + captioning module [1]	4.82	-	17.29
Bi-SST + captioning module [6]	9.60	19.10	12.68
Learnt proposals + DVC [18]	6.93	12.61	-
Learnt proposals + SDVC [32]	8.82	-	30.68
Learnt proposals + MDVC [33]	7.31	-	-
Our approach	**10.50**	**18.95**	**13.04**

Approaches	**BLEU-1**	**BLEU-2**	**BLEU-3**	**BLEU-4**
Bi-SST + H [6]	17.25	6.48	2.68	1.20
Bi-SST + E [6]	17.51	7.17	3.08	1.32
GT proposals + H-RNN [5]	19.46	8.78	4.34	2.53
learnt proposals + captioning module [1]	17.95	7.69	3.86	2.20
Bi-SST + captioning module [6]	18.99	8.84	4.41	2.30
Learnt proposals + DVC [18]	12.22	5.72	2.27	0.73
Learnt proposals + SDVC [32]	17.92	7.99	2.94	0.93
Learnt proposals + MDVC [33]	-	-	2.60	1.07
Our approach	**19.50**	**8.78**	**4.51**	**2.73**

Compared with the results of the first two reference approaches (*i.e.*, Bi-SST + H and Bi-SST + E), we can find that all evaluation scores can be improved by applying attention mechanisms instead of mean pooling for dynamically fusing video clip features and context vectors. This is because our approach can generate semantic-related words by paying more attention to visual features at each decoding time step.

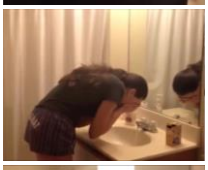
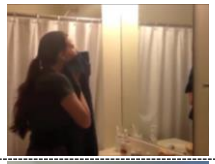
	GT	Ours
	A lady discusses and folds a towel.	A woman is talking to the camera.
	The lady washes her face with a powder from a box.	A woman washes her face and wipes her face with something.
	The lady rinses her face and use a towel to dry up.	A woman dry her face with a towel.
	Two girls are seen waving to the camera while one holds up a drink and the other plays with her hair.	Two girls are waving to the camera and talking to the camera.
	The girls walk around a bit followed by drinking out of a glass and making funny faces to the camera.	Two girls are walking around the camera and drinking in front of the camera.
	A man is playing cymbals under a bright light on a stage.	A man is drumming on the stage.
	He is joined by a woman on a bass drum.	Another person joins the man to play drums.
	Then they are joined by other drummers, and a man on a huge set in the background.	A group of people go on the stage to play drums.
	Lights flash as they perform in unison.	A group of people are drumming on the stage.

Figure 5.3. Qualitative examples of dense video captions generated by our approach.

Compared with the result of other state-of-the-art approaches, our proposed approach leverages the strength of RRNNs in temporal context extraction and relational reasoning to improve all evaluation scores. Therefore, our approach allows the

interactions (*i.e.*, fusion and reasoning) between past and future context information and the interactions between historical information across the video sequence. Figure 5.3 shows qualitative examples of dense video captions generated by our approach. We compared the generated results with the ground truths (GT).

In summary, the proposed approach can capture all events in the video, reason the relationships between these events, and then generate a set of informative sentences to describe the video content. Experimental results demonstrate that the proposed approach is superior to the state-of-the-art approaches on the ActivityNet Captions dataset.

5.5 Summary

Dense video captioning aims to locate and describe all events in a video to generate a set of sentences describing the video content. This emerging task in the field of digital video processing can be used to solve real-world problems, such as explaining a movie plot to the blind. Currently, various dense video captioning approaches have been developed. However, dense captions generated by these approaches are still not comparable to human descriptions.

This chapter proposes a novel Bidirectional Relational Recurrent Neural Network (Bidirectional RRNN) for dense video captioning, which takes advantage of context information to improve the quality of dense captions. The proposed approach collects and reasons temporal context information using a relational memory core, and thus can capture all events in the video, reason the relationships between these events, and then generate a set of informative sentences to describe the video content.

We test the proposed approach on the ActivityNet Captions dataset. Experimental results demonstrate that the proposed approach is superior to the state-of-the-art approaches. Our research verifies the effectiveness of the context information for generating high-quality dense video captions.

The proposed approach also can be improved. For example, we simply use multiplication for context fusion rather than developing a new fusion strategy. For future work, we intend to develop better dense video captioning approaches and more appropriate context fusion strategies for dense caption generation. We will also explore how to use other information such as audio and semantic information in generating dense video captions. In addition, we intend to apply dense video

captioning to a wider application field for solving real-world problems.

References

1. R. Krishna, K. Hata, F. Ren, F. Li, and J. C. Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 706-715. 2017.
2. D. L. Chen, and W. B. Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 190-200. 2011.
3. S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko. Translating videos to natural language using deep recurrent neural networks. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1494-1504. 2015.
4. S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4534-4542. 2015.
5. H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu. Video paragraph captioning using hierarchical recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4584-4593. 2016.
6. J. Wang, W. Jiang, L. Ma, W. Liu, and Y. Xu. Bidirectional attentive fusion with context gating for dense video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7190-7198. 2018.
7. S. Chen, T. Yao, and Y. Jiang. Deep learning for video captioning: a review. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pp. 6283-6290. 2019.
8. S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, S. Venugopalan, R. Mooney, T. Darrell, and K. Saenko. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2712-2719. 2013.
9. A. Kojima, T. Tamura, and K. Fukunaga. Natural language description of human activities from video images based on concept hierarchy of actions. *International Journal of Computer Vision*, 50(2): 171-184. 2002.
10. M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele. Translating video content to natural language descriptions. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 433-440. 2013.
11. R. Xu, C. Xiong, W. Chen, and J. J. Corso. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *the 29th*

AAAI Conference on Artificial Intelligence. pp. 2346-2352. 2015.

12. Z. Wu, T. Yao, Y. Fu, and Y. Jiang. Deep learning for video classification and captioning. In *Frontiers of Multimedia Research*, pp. 3-29. 2017.
13. C. Zhang, and Y. Tian. Automatic video description generation via LSTM with joint two-stream encoding. In *the 23rd International Conference on Pattern Recognition*, pp. 2924-2929. 2016.
14. L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. Describing videos by exploiting temporal structure. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4507-4515. 2015.
15. Z. Shen, J. Li, Z. Su, M. Li, Y. Chen, Y. Jiang, and X. Xue. Weakly supervised dense video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1916-1924. 2017.
16. F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 961-970. 2015.
17. V. Escorcia, F. C. Heilbron, J. C. Niebles, and B. Ghanem. Daps: Deep action proposals for action understanding. In *European Conference on Computer Vision*, pp. 768-784. 2016.
18. Y. Li, T. Yao, Y. Pan, H. Chao, and T. Mei. Jointly localizing and describing events for dense video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7492-7500. 2018.
19. S. Buch, V. Escorcia, C. Shen, B. Ghanem, and J. C. Niebles. SST: Single-stream temporal action proposals. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2911-2920. 2017.
20. X. Duan, W. Huang, C. Gan, J. Wang, W. Zhu, and J. Huang. Weakly supervised dense event captioning in videos. In *Advances in Neural Information Processing Systems*, pp. 3059-3069. 2018.
21. D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4489-4497. 2015.
22. A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and F. Li. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1725-1732. 2014.
23. A. Santoro, R. Faulkner, D. Raposo, J. Rae, M. Chrzanowski, T. Weber, D. Wierstra, O. Vinyals, R. Pascanu, and T. Lillicrap. Relational recurrent neural networks. In *Advances in Neural Information Processing Systems*, pp. 7299-7310. 2018.
24. J. Ma, R. Wang, W. Ji, H. Zheng, E. Zhu, and J. Yin. Relational recurrent neural

networks for polyphonic sound event detection. *Multimedia Tools and Applications*, 78(20): 29509-29527. 2019.

25. O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3156-3164. 2015.

26. S. Banerjee, and A. Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65-72. 2005.

27. K. Papineni, S. Roukos, T. Ward, and W. Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 311-318. 2002.

28. C. Lin. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74-81. 2004.

29. R. Vedantam, C. L. Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4566-4575. 2015.

30. X. Chen, H. Fang, T. Lin, R. Vedantam, S. Gupta, P. Dollár, and C. L. Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*. 2015.

31. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin *et al*. Tensorflow: A system for large-scale machine learning. In *the 12th Symposium on Operating Systems Design and Implementation*, pp. 265-283. 2016.

32. J. Mun, L. Yang, Z. Ren, N. Xu, and B. Han. Streamlined dense video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6588-6597. 2019.

33. V. Iashin, and E. Rahtu. Multi-modal dense video captioning. *arXiv preprint arXiv:2003.07758*. 2020.

34. R. Rothe, M. Guillaumin, and L. V. Gool. Non-maximum suppression for object detection by passing messages between windows. In *Asian Conference on Computer Vision*, pp. 290-306. 2014.

Chapter 6 Summary and future works

This chapter is the final chapter of this thesis. We firstly summarize each chapter and highlights the major contributions of this thesis in Section 6.1. Then the future works of this thesis are discussed in Section 6.2 for continuous research.

6.1 Research overview and summary

Multimedia data processing is a blooming and fast-growing research field, which covers a vast of diverse sub-fields and real-world applications. Digital audio processing and digital video processing are two important subfields of multimedia data processing. In this thesis, four novel approaches are proposed to address two key issues in multimedia data processing, *i.e*., (i) how to reduce the annotation costs of sound event classification/tagging, and (ii) how to improve the quality of video captions. Each proposed approach is presented specifically in the corresponding chapter.

Chapter 2 proposes a Gabor dictionary-based active learning (DBAL) approach for semi-supervised sound event classification, which addresses the issue of how to reduce the annotation costs of sound event classification/tagging. The proposed approach utilizes a Gabor dictionary for sound feature extraction. Then an active learning mechanism is used to select sound segments to be tagged with their true labels or predicted labels. After that, a sound event classifier can be trained using these recordings with their true or predicted labels, and the accuracy of the classifier is used to evaluate the performance of DBAL. Experimental results show that DBAL achieves comparable classification accuracy but requires fewer annotation costs compared with other existing semi-automatic sound event classification approaches.

To deal with the same issue, Chapter 3 further proposes a learnt dictionary-based active learning (LDAL) approach for semi-supervised sound event tagging. Compared with DBAL, LDAL utilizes a K-SVD learnt dictionary to replace the Gabor dictionary for sound feature extraction. Both LDAL and DBAL use the same way for actively labeling and sound event classification. Then the tagging accuracy and annotation cost are used to measure the performance of LDAL. We test LDAL on two public urban sound datasets. Experimental results show that LDAL achieves higher tagging accuracy but requires much fewer annotation costs than other

reference approaches.

Chapter 4 proposes a novel attention-based dual learning (ADL) approach for video captioning, which addresses the issue of how to improve the quality of video captions. Specifically, ADL consists of two modules: a caption generation module and a video reconstruction module. The encoder-decoder-reconstructer structure conducted by the two modules allows ADL to leverage the information from both raw videos and generated captions for video caption generation. Moreover, a multi-head attention mechanism is utilized in both two modules, helping the two modules focus on the most effective information in raw videos and generated captions, and a dual learning mechanism is used to fine-tune the two modules. Thus, ADL can minimize the semantic gap between raw videos and generated captions by minimizing the differences between the reproduced and raw videos. Experimental results on benchmark datasets demonstrated that ADL can generate high-quality video captions.

To deal with the same issue, Chapter 5 further proposes a novel bidirectional relational recurrent neural network (Bidirectional RRNN) for dense video captioning. Different from ADL, which is used to generate a single-sentence caption to describe the main content of a video, Bidirectional RRNN can generate dense video captions (*i.e.*, a set of sentences) for the video containing multiple video events. In Bidirectional RRNN, a bidirectional RRNN encoder is used to obtain the local and global context information of a target event. Thus, the proposed approach can capture all events in videos, reason the relationships between these events, and then generate a set of informative sentences to describe video contents. Experimental results demonstrate that the proposed approach is superior to the state-of-the-art approaches on the ActivityNet Captions dataset.

In summary, this thesis proposes four effective approaches for processing multimedia data, which address two key issues in multimedia data processing, *i.e.*, (i) how to reduce the annotation costs of sound event classification/tagging, and (ii) how to improve the quality of video captions. Experimental results show that our approaches outperform the state-of-the-art approaches.

6.2 Future work

This thesis proposes four effective approaches for multimedia data processing. However, the potential of these developed approaches has not been fully explored. In this section, several research directions are discussed to extend the potential of

these approaches.

The first research direction is the neural architecture search (NAS), which aims to automatically search for optimal neural architectures for different deep neural network-based tasks [1]. Recently, NAS-based neural architectures have outperformed manually designed neural architectures on some simple tasks such as image classification and semantic segmentation [1-6]. In this thesis, the approaches proposed in Chapters 4 and 5 are based on manually designed deep neural network architectures. Therefore, the first future research direction is how to use existing NAS approaches or develop new NAS approaches to optimize the neural architecture of video captioning approaches.

The second research direction is multimodal learning-based video captioning, which aims to build models that can relate and process information from multiple modalities in a video, such as audio, frame, and motion modalities, for caption generation [7]. Multimodal learning-based video captioning approaches can capture the correspondence between different modalities and provide the possibility of an in-depth understanding of the video content. In this thesis, the approaches proposed in Chapters 2 and 3 are related to sound data processing, and the approaches proposed in Chapters 4 and 5 are related to sequence learning-based video captioning. Therefore, the second future research direction is how to develop multimodal learning-based approaches for video captioning, which can integrate information from different modalities in videos for caption generation. The four new approaches proposed in this thesis will be a help to the exploration in this direction.

In summary, the approaches proposed in this thesis can be further extended to many research directions. For future work, we intend to develop new NAS approaches to optimize the neural architectures for video captioning. We will also explore how to integrate multimodal information in videos for video caption generation. In addition, we intend to apply video captioning approaches to a wider application field for solving real-world problems.

References

1. L. Li, and A. Talwalkar (2019) Random search and reproducibility for neural architecture search. In *the 6th ICML Workshop on Automated Machine Learning*, pp.1-20.
2. K. Kandasamy, W. Neiswanger, J. Schneider, B. Poczos, and E. P. Xing (2018) Neural

architecture search with Bayesian optimization and optimal transport. In *Advances in Neural Information Processing Systems*, pp. 2016-2025.

3. H. Liu, K. Simonyan, and Y. Yang (2019) Darts: Differentiable architecture search. In *the International Conference on Learning Representations*, pp. 1-13.
4. R. Luo, F. Tian, T. Qin, E. Chen, and T. Liu (2018) Neural architecture optimization. In *Advances in Neural Information Processing Systems*, pp. 7827-7838.
5. C. Gao, Y. Chen, S. Liu, Z. Tan, and S. Yan (2020) AdversarialNAS: Adversarial neural architecture search for GANs. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-10.
6. A. Gaier, and D. Ha (2019) Weight agnostic neural networks. In *Advances in Neural Information Processing Systems*, pp. 1-15.
7. T. Baltrušaitis, C. Ahuja, and L. P. Morency. Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2): 423-443. 2018.

Appendix A list of publications

1. **Wanting Ji**, Ruili Wang, and Junbo Ma. Dictionary-based active learning for sound event classification. *Multimedia Tools and Applications*, 78(3): 3831-3842. 2019.
2. **Wanting Ji**, Ruili Wang, Yan Tian, and Xun Wang. An attention based dual learning approach for video captioning. *Neurocomputing*, submitted.
3. **Wanting Ji**, Ruili Wang, and Mingzhe Liu. Dense video captioning with context fusion and reasoning. *Image and Vision Computing*, submitted.
4. Xiao Qin, **Wanting Ji** (corresponding author), Ruili Wang, and ChangAn Yuan. Learnt dictionary based active learning method for environmental sound event tagging. *Multimedia Tools and Applications*, 78(20): 29493-29508. 2019.

MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

GRADUATE RESEARCH SCHOOL

STATEMENT OF CONTRIBUTION
DOCTORATE WITH PUBLICATIONS/MANUSCRIPTS

We, the candidate and the candidate's Primary Supervisor, certify that all co-authors have consented to their work being included in the thesis and they have accepted the candidate's contribution as indicated below in the *Statement of Originality*.

Name of candidate:	Wanting Ji
Name/title of Primary Supervisor:	Professor Ruili Wang
In which chapter is the manuscript /published work:	Chapter 2

Please select one of the following three options:

The manuscript/published work is published or in press

- Please provide the full reference of the Research Output:

Wanting Ji, Ruili Wang, and Junbo Ma (2019) Dictionary-based active learning for sound event classification. Multimedia tools and applications, 78(3): 3831-3842.

The manuscript is currently under review for publication – please indicate:

- The name of the journal:

- The percentage of the manuscript/published work that was contributed by the candidate:

- Describe the contribution that the candidate has made to the manuscript/published work:

It is intended that the manuscript will be published, but it has not yet been submitted to a journal

Candidate's Signature:	Wanting Ji 数字签名者：Wanting Ji 日期：2020.06.28 14:17:52 +12'00'
Date:	28-Jun-2020
Primary Supervisor's Signature:	Prof Ruili Wang Digitally signed by Prof Ruili Wang Date: 2020.06.30 12:13:36 +12'00'
Date:	30-Jun-2020

This form should appear at the end of each thesis chapter/section/appendix submitted as a manuscript/ publication or collected as an appendix at the end of the thesis.

MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

GRADUATE RESEARCH SCHOOL

STATEMENT OF CONTRIBUTION
DOCTORATE WITH PUBLICATIONS/MANUSCRIPTS

We, the candidate and the candidate's Primary Supervisor, certify that all co-authors have consented to their work being included in the thesis and they have accepted the candidate's contribution as indicated below in the *Statement of Originality*.

Name of candidate:	Wanting Ji
Name/title of Primary Supervisor:	Professor Ruili Wang
In which chapter is the manuscript /published work:	Chapter 3

Please select one of the following three options:

The manuscript/published work is published or in press

- Please provide the full reference of the Research Output:

Xiao Qin, Wanting Ji (corresponding author), Ruili Wang, and ChangAn Yuan (2019) Learnt dictionary based active learning method for environmental sound event tagging. Multimedia Tools and Applications, 78(20): 29493-29508.

The manuscript is currently under review for publication – please indicate:

- The name of the journal:

- The percentage of the manuscript/published work that was contributed by the candidate:

- Describe the contribution that the candidate has made to the manuscript/published work:

It is intended that the manuscript will be published, but it has not yet been submitted to a journal

Candidate's Signature:	Wanting Ji 数字签名者：Wanting Ji 日期：2020.06.28 14:22:41 +12'00'
Date:	28-Jun-2020
Primary Supervisor's Signature:	Prof Ruili Wang Digitally signed by Prof Ruili Wang Date: 2020.06.30 12:15:15 +12'00'
Date:	30-Jun-2020

This form should appear at the end of each thesis chapter/section/appendix submitted as a manuscript/ publication or collected as an appendix at the end of the thesis.

MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

GRADUATE RESEARCH SCHOOL

STATEMENT OF CONTRIBUTION
DOCTORATE WITH PUBLICATIONS/MANUSCRIPTS

We, the candidate and the candidate's Primary Supervisor, certify that all co-authors have consented to their work being included in the thesis and they have accepted the candidate's contribution as indicated below in the *Statement of Originality*.

Name of candidate:	Wanting Ji
Name/title of Primary Supervisor:	Professor Ruili Wang
In which chapter is the manuscript /published work:	Chapter 4

Please select one of the following three options:

The manuscript/published work is published or in press

- Please provide the full reference of the Research Output:

The manuscript is currently under review for publication – please indicate:

- The name of the journal:

Neurocomputing

- The percentage of the manuscript/published work that was contributed by the candidate: 70.00

- Describe the contribution that the candidate has made to the manuscript/published work:

For this manuscript, the contributions that the candidate has made include conceptualization, investigation, methodology, validation, and draft writing.

It is intended that the manuscript will be published, but it has not yet been submitted to a journal

Candidate's Signature:	Wanting Ji 数字签名者：Wanting Ji 日期：2020.06.28 14:30:04 +12'00'
Date:	28-Jun-2020
Primary Supervisor's Signature:	Prof Ruili Wang Digitally signed by Prof Ruili Wang Date: 2020.06.30 12:16:26 +12'00'
Date:	30-Jun-2020

This form should appear at the end of each thesis chapter/section/appendix submitted as a manuscript/ publication or collected as an appendix at the end of the thesis.

MASSEY UNIVERSITY
TE KUNENGA KI PŪREHUROA
UNIVERSITY OF NEW ZEALAND

GRADUATE RESEARCH SCHOOL

STATEMENT OF CONTRIBUTION
DOCTORATE WITH PUBLICATIONS/MANUSCRIPTS

We, the candidate and the candidate's Primary Supervisor, certify that all co-authors have consented to their work being included in the thesis and they have accepted the candidate's contribution as indicated below in the *Statement of Originality*.

Name of candidate:	Wanting Ji
Name/title of Primary Supervisor:	Professor Ruili Wang
In which chapter is the manuscript /published work:	Chapter 5

Please select one of the following three options:

The manuscript/published work is published or in press

- Please provide the full reference of the Research Output:

The manuscript is currently under review for publication – please indicate:

- The name of the journal:

IEEE Transactions on Circuits and Systems for Video Technology

- The percentage of the manuscript/published work that was contributed by the candidate: 70.00

- Describe the contribution that the candidate has made to the manuscript/published work:

For this manuscript, the contributions that the candidate has made include conceptualization, investigation, methodology, validation, and draft writing.

It is intended that the manuscript will be published, but it has not yet been submitted to a journal

Candidate's Signature:	Wanting Ji 数字签名者：Wanting Ji 日期：2020.06.28 14:32:49 +12'00'
Date:	28-Jun-2020
Primary Supervisor's Signature:	Prof Ruili Wang Digitally signed by Prof Ruili Wang Date: 2020.06.30 12:17:27 +12'00'
Date:	30-Jun-2020

This form should appear at the end of each thesis chapter/section/appendix submitted as a manuscript/ publication or collected as an appendix at the end of the thesis.