



Doctoral Thesis

A Study on High-Level Cognitive Understanding of Images towards Language

Bei LIU

September 2018

Department of Social Informatics
Graduate School of Informatics
Kyoto University

Doctoral Thesis
submitted to Department of Social Informatics,
Graduate School of Informatics,
Kyoto University
in partial fulfillment of the requirements for the degree of
DOCTOR of INFORMATICS

Thesis Committee: Masatoshi Yoshikawa, Professor
Takayuki Kanda, Professor
Shinsuke Mori, Professor

A Study on High-Level Cognitive Understanding of Images towards Language*

Bei LIU

Abstract

Nowadays, a large number of images are flooding our local memory and social image sharing websites as a result of camera and smart phone's popularization. Images are becoming a more used way for expression compared with past days. Meanwhile, language, as another important form of communication, is also attracting our attention for research. The interaction between image and language is not unexplored yet though we humans perform so many tasks that involve both modality. Though low-level cognitive (e.g. facts) understanding of images is largely tackled and has achieved great success, high-level cognitive understanding of images still remains a challenge.

In this research, we explore the importance of high-level cognitive understanding towards language from two types of tasks: search-based problems and generation-based problems. Different forms of languages are involved in these tasks, including words of event, words of subjective adjective, stories and poems.

To bridge images and events, we tackle the problem of event summarization from images, which aims to retrieve images to represent an event with high perceptual quality. Instead of directly searching for related images of a certain event, we propose to find images that cannot misrecognized as its neighbor events, which we define three types, namely sub-event, super-event and sibling-event. We analyze the reasons of these misrecognitions and propose method to prevent from them accordingly.

In the research of learning subjective adjectives from images, we propose to distinguish relevant and irrelevant images in weakly-labeled data with a pairwise stacked convolutional auto-encoder that can learn discriminative features by identifying a dominant difference between them. We define pseudo-relevant and pseudo-irrelevant image sets as results obtained from image search engines with query with or without the subjective adjective.

*Doctoral Thesis, Department of Social Informatics, Graduate School of Informatics, Kyoto University, KU-I-DT6960-26-0400, September 2018.

To generate a story from a sequence of images towards human cognition, we take emotion as an important factor that guides the generation of a story. The task is formulated into two correlated tasks: generating sentences based on both visual contents and emotions, and predicting possible emotions from images considering contextual images in a sequence. An emotion conditioned story generation model is proposed to guide image embedding learning and story decoder, while a RNN-based prediction model is proposed to learn emotions of each image in a sequence considering contextual images.

The task of poetry generation is challenging as writing a poem involves multiple principles. The difficulties mainly drive from discovering the poetic clues from an image (e.g., rose for love), and generating poems to satisfy both relevance to images and the poeticness in language. We formulate the task of poem generation into two correlated sub-tasks by multi-adversarial training via policy gradient, through which the cross-modal relevance and poetic language style can be ensured. To convey the poetic clues from poems, we propose to learn a deep coupled visual-poetic embedding, in which the poetic representation for objects, sentiments and scenes from images can be jointly learned. Two discriminative networks are further introduced to guide the poem generation, including a multi-modal discriminator and a poem-style discriminator.

To draw a brief conclusion, the work presented in this thesis have made the following progress. (1) We proposed to analyze and understand images from the high-level cognitive perspective; (2) We devised novel models and algorithms from two types of tasks: search-based and generation-based; (3) Our work were verified with extensive experiments. Encouraging results were obtained by comparison with state-of-the-art baselines.

Keywords: Image, Language, High-Level, Cognitive Understanding.

CONTENTS

1	Introduction	1
1.1	Background	1
1.1.1	Image	1
1.1.2	Language	2
1.1.3	High-Level Cognitive Understanding of Images towards Lan- guage	3
1.2	Overview of the Research	4
1.3	Motivation and Tasks	5
1.3.1	Event Summarization with Images	5
1.3.2	Learning Subjective Adjectives from Images	6
1.3.3	Visual Storytelling	6
1.3.4	Poetry Generation from Image	6
1.4	Thesis Structure	7
2	Technical Preliminaries	9
2.1	Maximal Marginal Relevance	9
2.2	Auto-Encoder	10
2.2.1	Denoising Auto-Encoder	10
2.2.2	Convolutional Auto-Encoder	11
2.2.3	Stacked Auto-Encoder	11
2.3	Recurrent Neural Network	11
2.4	Adversarial Training	12
2.5	Reinforcement Learning	14

3	Event Summarization with Images	16
3.1	Introduction	16
3.2	Related Work	18
3.2.1	Event Representation	19
3.2.2	Image Summarization	19
3.3	Preliminaries	20
3.3.1	Events	20
3.3.2	Problem Definition	21
3.4	Approach	22
3.4.1	Overview	22
3.4.2	Misrecognition Situations	22
3.4.3	Sub-Event Coverage	24
3.4.4	Super-Event Coverage	26
3.4.5	Difference from Sibling-Event	27
3.4.6	Image Set Generation	29
3.5	Neighbor Event Generation	29
3.5.1	Generating Sub-Events	30
3.5.2	Generating Super-Events	32
3.5.3	Generating Sibling-Events	32
3.6	Experiments	34
3.6.1	Generating Neighbor Events	34
3.6.2	Generating Image Sets	37
3.6.3	Analysis of Events and Perceptual Quality	44
3.6.4	Image Set Size Selection	44
3.7	Conclusion and Future Work	47
4	Learning Subjective Adjectives from Images	49
4.1	Introduction	49
4.2	Related Work	52
4.3	Preliminaries	52
4.3.1	Terminology Definitions	52
4.3.2	Problem Definition	53
4.4	Approach	54
4.4.1	Image Pair Construction	55
4.4.2	Pair-Wise Stacked Convolutional Auto-Encoder	56

Contents

4.4.3	Image Ranking with Learnt Features	59
4.5	Experiment	59
4.5.1	Datasets	60
4.5.2	Network Structure	61
4.5.3	Result	62
4.6	Conclusion	63
5	Visual Storytelling	65
5.1	Introduction	65
5.2	Related Work	68
5.2.1	Visual Description Generation	68
5.2.2	Visual Storytelling	68
5.3	Approach	69
5.3.1	Overview	69
5.3.2	Emotion Generator	71
5.3.3	Story Generator	71
5.3.4	Emotion Feature	72
5.3.5	Decoder	72
5.3.6	Reinforcement Learning	72
5.4	Experiment	72
5.4.1	Dataset and Analysis	72
5.4.2	Experiment Settings	74
5.4.3	Compared Methods	74
5.4.4	Objective Evaluation Metrics	74
5.4.5	Results and Analysis	75
5.5	Conclusion	75
6	Poetry Generation from Image	78
6.1	Introduction	78
6.2	Related Work	81
6.2.1	Poetry Generation	81
6.2.2	Image Description	81
6.3	Approach	83
6.3.1	Deep Coupled Visual-Poetic Embedding	83
6.3.2	Poem Generator as an Agent	85
6.3.3	Discriminators as Rewards	86

Contents

6.3.4	Multi-Adversarial Training	88
6.4	Experiments	89
6.4.1	Datasets	89
6.4.2	Compared Methods	90
6.4.3	Automatic Evaluation Metrics	91
6.4.4	Human Evaluation	92
6.4.5	Training Details	93
6.4.6	Evaluations	93
6.5	Conclusion and Discussion	99
7	Conclusion and Future Work	101
7.1	Conclusion	101
7.2	Future Work	102
	Acknowledgements	105
	References	107
	Selected List of Publications	120
	Appendix	122
A	Proof for Section 3.4.6	122
A.1	Proof of Submodularity	122
A.2	Proof of Monotonicity	123

LIST OF FIGURES

1.1	Framework of Doctoral Thesis.	4
1.2	Tasks in terms of method.	5
2.1	The denoising auto-encoder architecture.	10
2.2	Illustration of (a) LSTM and (b) GRU. (a) i , f and o are the input, forget and output gates, respectively. c and $\tilde{c}$ denote the memory cell and the new memory cell content. (b) r and z are the reset and update gates, and h and $\tilde{h}$ are the activation and the candidate activation. . .	12
2.3	An illustration of generative adversarial framework. The generative model G takes as input a noise vector z and generate an image of human face, while the discriminative model (D) first learns the difference between the real face images and the generated images and then optimize G by switching the label of the generated samples from 0 to 1. .	13
2.4	An illustration of reinforcement learning system.	14
3.1	Example of Sub-Event Coverage: image set with only images of “Kyoto” (B) will indicate “Kyoto travel spring Japan”, which is a sub-event of “Japan spring travel”, while an image set that covers different parts of Japan (A) will not have this problem.	25
3.2	Example of Super-Event Coverage: image set with only images of “Kyoto” (B) will indicate “Kyoto”, which is a super-event of “Kyoto travel”, while an image set that covers both “Kyoto” and “travel” (A) will not have this problem.	27

List of Figures

3.3	Example of difference from sibling-events. Images of willow trees are common in both Japan and China in spring, so an image set that includes an image of a willow tree (A) is not as good as one that includes an image of women in kimono (B), since an image of women in kimono is unique to Japan and will not cause misrecognition. . . .	28
3.4	Neighbor Events of “travel Japan spring”	35
3.5	Value distribution of three criteria for event “travel Japan spring”. . .	38
3.6	Image sets of event “walk dog” generated by different methods: VR (baseline VisualRank), SibDif (difference from sibling-events), and ALL (combination of three criteria).	40
3.7	Perceptual quality of five methods: we applied each method to 50 events in five different sizes of image sets: 1, 3, 5, 7, and 9.	41
3.8	Perceptual quality of several events in different sizes.	42
3.9	Perceptual quality of different types of events and methods. Here, “None” corresponds to non-time specified or non-location specified, “time” to time-specified, “location” to location-specified, and “time & location” to time specified and location specified.	45
3.10	Evaluation of image set size selection: sizes generated in automatically (yellow dots) and sizes selected manually by considering each event’s perceptual quality and smallness of size(blue dots).	46
4.1	Two relevant images with the same “subjective adjective” but different “nouns”.	50
4.2	The framework of pair-wise stacked convolutional auto-encoder architecture.	56
4.3	Structure of neural network in experiment.	61
5.1	Example of stories annotated by different users for the same image. For image (b), annotator 1 reads surprising from the kid’s face while annotator captures excitement . For image (d), the kids are excited for annotator 1 while they are giddy for annotator 2. Sentences for these two images contain different contents with different emotion interpretation.	66
5.2	Framework of our approach.	70

List of Figures

5.3	Distribution of emotions in the training dataset of VIST. Emotions are extracted by DeepMoji and clustered into 8 types, in which a typical emoji is shown.	73
5.4	Example of stories generated by state-of-art method [1] and our approach.	76
6.1	Example of human written description and poem of the same image. We can see a significant difference from words of the same color in these two forms. Instead of describing facts in the image, poem tends to capture deeper meaning and poetic symbols from objects, scenes and sentiments from the image (such as knight from <i>falcon</i> , hunting and fight from <i>eating</i> , and waiting from <i>standing</i>).	79
6.2	The framework of poetry generation with multi-adversarial training. A deep coupled visual-poetic model (e) is trained by human annotated image-poem pairs (a). The image features (b) are poetic multi-CNN features obtained by fine-tuning CNNs with the extracted poetic symbols (e.g., objects, scenes and sentiments) by a POS parser [2] from poems. The sentence features (d) of poems are extracted from a skip-thought model (c) trained on the largest public poem corpus (UniM-Poem). A RNN-based sentence generator (f) is trained as agent and two discriminators considering multi-modal (g) and poem-style (h) critics of a generated poem to a given image provide rewards to policy gradient (i). POS parser extracts Part-Of-Speech words from poems. .	82
6.3	Examples in two datasets: UniM-Poem and MultiM-Poem.	89
6.4	Example of poems generated by eight methods for an image. Words in read indicate poeticness.	95
6.5	Example of poems generated by our approach I2P-GAN.	96
6.6	Automatic evaluation. Note that BLEU scores are computed in comparison with human-annotated ground-truth poems (one poem for one image). Overall score is computed as an average of three metrics after normalization (Eq. (6.16)). All scores are reported as percentage (%).	97

LIST OF TABLES

3.1	50 events categorized based on whether the event is time-aware or location-aware	36
3.2	Comparison of perceptual quality of image sets generated by our proposed method and TF-IDF method.	43
4.1	The queries we used in the experiment. (Ratio A: ratio of truly relevant images in pseudo-relevant images, Ratio B: ratio of truly relevant images in pseudo-irrelevant images)	60
4.2	Result of our approach and the precision of top 200 in image search engines for two queries.	62
5.1	Automatic evaluation. All scores are reported as percentage (%). . . .	75
6.1	Detailed information about the three datasets. The first two datasets are collected by ourselves and the third one is extended by our embedding model.	90
6.2	Average score of relevance to images for three types of human written poems on 0-10 scale (0-irrelevant, 10-relevant). One-way ANOVA revealed that evaluation on these poems is statistically significant ($F(2,9) = 130.58, p < 1e - 10$).	94
6.3	Human evaluation results of six methods on four criteria: relevance (Rel), coherence (Coh), imaginativeness (Imag) and Overall. All criteria are evaluated on 0-10 scale (0-bad, 10-good).	98
6.4	Accuracy of Turing test on AMT users and expert users on poems with and without images.	98

CHAPTER 1

INTRODUCTION

In this chapter, we first introduce the background of this doctor thesis from both the social perspective and driving by technological development. Then, we will have an overview of this research. Following that, we introduce tasks included in this thesis and their motivations. We show the structure of this thesis in the last section.

1.1 Background

1.1.1 Image

Images are used all around the world today. With the widely use of digital camera and smart phone, everyone becomes the master of image at anytime in anywhere. Images have become a way of expression and a resource of inspiration. From the perspective of image creators and users, images are taken and applied for different purpose. Some images are used to transfer information and show facts, such as images reported in some news and shown in some presentations. Some images are created to promote viewers feelings, like encouraging images used by many public account in social network and images for recreation. Some images are supposed to motivate user behavior, such as promotional images used for political election and advertisement. There are also some images taken only for recoding, sharing and memory.

A picture is worth a thousand words, as the saying goes, most images play a role of communication, whether they are used for fact, feeling promotion or behavior motivation. Image-based communication has advantages in terms of its efficiency (less time necessary for understanding the content), perspicuity (able to view more contents at the same time), and language-independence (comprehensible for speakers of any language), while it cannot be correctly understood unless appropriate images are used for the communication.

In the field of computer science, images can be studied for many years. Recently, success of deep learning technologies has brought the understanding of images by computers to a human level. Image related tasks, such as image segmentation, object recognition, object detection, scene recognition *etc.* have achieved approaching performance or even outperform results. However, from human's perspective, those tasks are about facts of the images and more related to human recognition or low-level cognitive understanding. How to make computer understand images in a high-level cognitive way still remains a big challenge. Tasks that are related to high-level cognitive understanding of images include emotion prediction, aesthetic estimation, *etc.*

1.1.2 Language

When we talk about communication, another important word comes to our mind might be "language". As a matter of fact, when you are reading these words in this thesis, you are taking part in one of the wonders of the natural world [3]. Language is an aspect of human behavior and it is a set of symbols being used for communication. Different types of language exist, including spoken language, written language. In this thesis, when we mention language, we refer to it as written language. Among written languages, words, sentences, paragraphs are all forms of it.

Also benefiting from deep learning, we have gained deeper understanding of language recent year. From the perspective of perception, we can cast the understanding of languages into three stages. We first learn how to use languages to represent. In this stage, we get to know "dog" and "cat" both belong to "animal" and "travel in Kyoto" is similar to "travel to Tokyo". Then we learn to use language in tasks related to machine intelligence, such as question and answer, conversation, *etc.* The highest stage is to make machine cognitive by simulating human's expression for our cognitions, such as poem writing.

If we simulate computer's understanding of images as a process of human percep-

tion, we can imagine language understanding as a process of learning to write papers. We first learn to record the fact, then to write the reasoning and arguments. When we reach a certain level of writing standard, we will try to express ourselves with poem or prose.

1.1.3 High-Level Cognitive Understanding of Images towards Language

Most current image related tasks are focusing on images' low-level cognitive understanding, which infers recognition tasks of vision computing. We use the term **high-level cognitive understanding** of images to indicate image understanding in terms of human cognition, such as emotion, aesthetic or poetic inspirations.

The interaction between image and language is conducted very often in our daily life, even though we have not realized it. When we ask someone to imagine a picture in his/her mind about a situation we tell them, when we try to describe some situation you have seen with languages, we are performing a task that bridge the image and language.

There are some researches to bridge image and language. Search based technologies are first explored. Images and language (word or sentence) are first represented with handcrafted or deep learning features [4, 5]. Then ranking algorithms are applied to retrieve the most similar images for a linguistic expression or the most similar language for an image. Recently, generative technologies have promoted works to generate languages given images (image captioning [6], image paragraphing [7], visual storytelling [8, 1], visual question and answer [9], visual response generation [10]) or generate images given language (image generation given keywords or sentence [11]).

In this research, we focus on image understanding towards language, especially high-level cognitive understanding such as emotion, event, story and poetic inspirations that require human's understanding of images beyond fact recognition from images. Both search-based and generative methods are explored to mine the high-level cognitive concepts we can get from images and represent it in languages. We will explain the overview of this research in the next section.

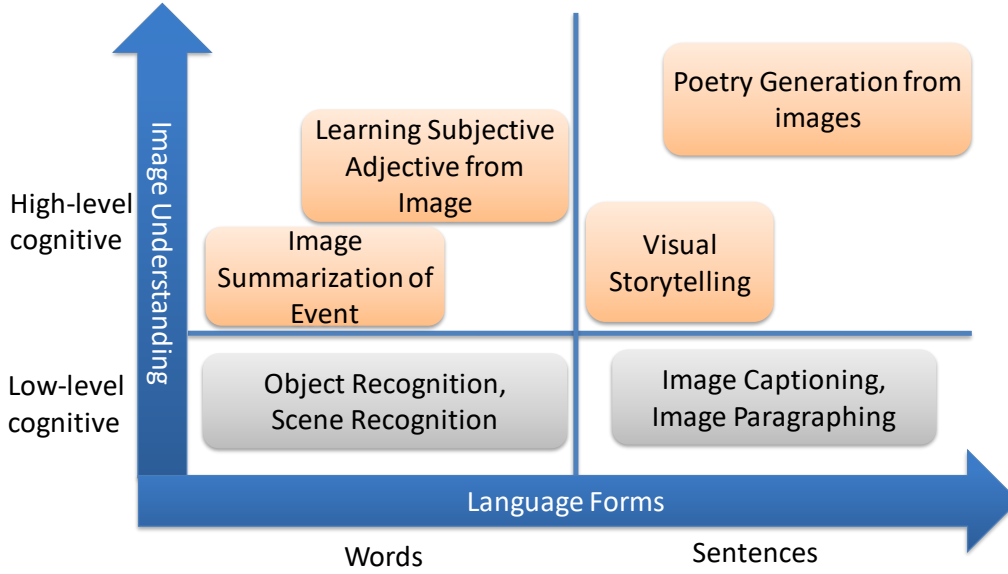


Figure 1.1. Framework of Doctoral Thesis.

1.2 Overview of the Research

The overall framework is shown in Figure 1.1. We use two dimensions to indicate the connection between our topics. One is based on whether our understanding of images is high-level cognitive or low-level cognitive. Many existing works have been done to low-level cognitive understanding of images, such as object recognition from images, scene recognition from images, image captioning and image paragraphing. This doctoral thesis will concentrate on high-level cognitive understanding of images. Another dimension classify language into two forms: words and sentences. We have two topics concerning words while another two concerning sentences.

From the perspective of methods, our tasks can be distributed as in Figure 1.2. Event summarization with images aims to search for images to present an event, and learning subjective adjectives from images targets to re-rank images given words of a subjective adjective to make the result more related. Visual storytelling tries to generate several sentences for a sequence of images to form a story, and poetry generation from image will automatically generate a poem given an image.

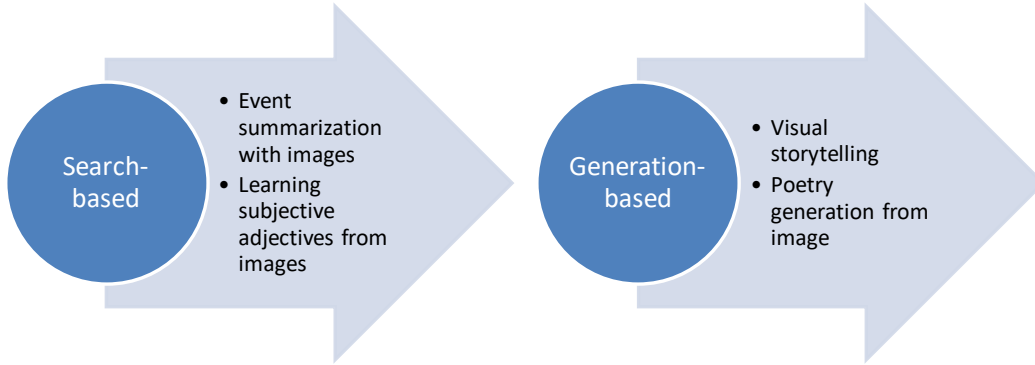


Figure 1.2. Tasks in terms of method.

1.3 Motivation and Tasks

Since detailed motivations will be explained in the following chapters respectively, in this section, we briefly introduce the motivation, target and approach of each task.

1.3.1 Event Summarization with Images

Using image summarization technology to summarize an image collection is often oriented to users who own this image collection. However, people’s interest of sharing images with others highlights the importance of *cognition-aware* summarization of images by which viewers can easily recognize the exact event those images represent. In this research, we address the problem of cognition-aware summarization of images representing events, and propose to solve this problem and to improve perceptual quality of an image set by proactively avoiding misrecognition that an image set might bring.

Three types of neighbor events that are possible to cause misrecognitions are discussed in this work, namely sub-event, super-event and sibling-event. We analyze the reasons of these misrecognitions and then propose three criteria to prevent from them accordingly. Combination of the criteria is used to generate summarization of images that can represent an event with several images.

Our approach is empirically demonstrated with images from Flickr by utilizing their visual features and related tags. Comparing with a baseline method, the result demonstrates the effectiveness of our proposed methods.

1.3.2 Learning Subjective Adjectives from Images

In this paper, we propose a method of learning subjective adjectives (i.e., adjectives that express opinions and evaluations in natural languages [12]) from images retrieved in image search engines with “subjective adjective noun pair” (ANP) queries. Since there are a variety of subjective adjective-noun pairs that can be used in ANP queries, we do not rely on labeled datasets that require a tremendous cost, but exploit results from existing image search engines as weakly-labeled data, which contain labels with a lot of errors.

To effectively distinguish relevant and irrelevant images in the weakly-labeled data, we propose a *pairwise stacked convolutional auto-encoder* that can learn discriminative features by identifying a dominant difference between two sets of images, namely, pseudo-relevant and pseudo-irrelevant image sets obtained from image search engines.

We conducted experiments with images from Flickr to evaluate the effectiveness of our approach, and found that the proposed approach could effectively learn subjective adjectives even without human-labeled data.

1.3.3 Visual Storytelling

Automatic generation of story from a sequence of images, i.e., visual storytelling, has attracted extensive attention. Existing works focus on story generation based on visual contents. However, even the same sequence of images will lead to different stories. In this work, we take emotion as the important factor that guides the generation of story.

The task is formulated into two correlated sub-tasks: generating sentences based on both visual contents and emotions, and predicting possible emotions from images considering contextual images in a sequence.

We first propose an emotion conditioned story generation model to guide image embedding learning and story decoder. Then we propose a RNN-based prediction model to learn emotions of each image in a sequence considering contextual images.

1.3.4 Poetry Generation from Image

Automatic generation of natural language from images has attracted extensive attention. In this work, we take one step further to investigate generation of poetic language (with multiple lines) to an image for automatic poetry creation.

This task involves multiple challenges, including discovering poetic clues from the

image (e.g., hope from green), and generating poems to satisfy both relevance to the image and poeticness in language level. To solve the above challenges, we formulate the task of poem generation into two correlated sub-tasks by multi-adversarial training via policy gradient, through which the cross-modal relevance and poetic language style can be ensured. To extract poetic clues from images, we propose to learn a deep coupled visual-poetic embedding, in which the poetic representation from objects, sentiments^{*} and scenes in an image can be jointly learned. Two discriminative networks are further introduced to guide the poem generation, including a multi-modal discriminator and a poem-style discriminator.

To facilitate the research, we have released two poem datasets by human annotators with two distinct properties: 1) the first human annotated image-to-poem pair dataset (with 8,292 pairs in total), and 2) to-date the largest public English poem corpus dataset (with 92,265 different poems in total). Extensive experiments are conducted with 8K images, among which 1.5K image are randomly picked for evaluation. Both objective and subjective evaluations show the superior performances against the state-of-the-art methods for poem generation from images. Turing test carried out with over 500 human subjects, among which 30 evaluators are poetry experts, demonstrates the effectiveness of our approach.

1.4 Thesis Structure

The structure of this thesis is as follows. In Chapter 2, a review of related works, especially some techniques that will be used in the following works, will be described. Chapter 3 introduces topic of event summarization with images by analyzing the relationship between events and proposing methods for image selection. Chapter 4 is about how to learning subjective adjectives from images by first defining subjective adjectives and our approach of pair-wise stacked convolutional auto-encoder. In Chapter 5, we explain the task of visual storytelling and our proposal of introducing emotion as an important factor for story generation from image sequence. Chapter 6 introduces the task of poetry generation from images and our approach of incorporating visual-poetic embedding and poem generation in adversarial way. Finally, Chapter 7 draws a conclusion of this thesis and has a discussion about the future researches.

^{*}We consider both adjectives and verbs that can express emotions and feelings as sentiment words in this research.

CHAPTER 2

TECHNICAL PRELIMINARIES

In this chapter, we will introduce some technical background we have used in our four researches.

2.1 Maximal Marginal Relevance

Maximal Marginal Relevance (MMR) is a criterion used in retrieval task and it aims to reduce redundancy while maintaining query relevance in re-ranking retrieved documents or images [13]. Marginal relevance is defined as linear combination of high relevance and minimal redundancy. A document is marked as high marginal relevance if it is relevant to the query and contains minimal similarity to previously selected documents. And we target to maximize marginal relevance for each document D_i in the unselected document set as follows:

$$\text{MMR} = \operatorname{argmax}_{D_i \in R \setminus S} \left[\lambda \text{Sim}_1(D_i, Q) - (1 - \lambda) \max_{D_j \in S} \text{Sim}_2(D_i, D_j) \right], \quad (2.1)$$

where Q is a query or user profile, S is the subset of documents in the already selected result list. $R \setminus S$ is the set difference, i.e., the set of as yet unselected documents. Sim_1 is the similarity metric used in the document retrieval and relevance ranking between documents and a query, and Sim_2 can be the same as Sim_1 or a different metric.

2.2 Auto-Encoder

Encoder-decoder paradigm is used in many unsupervised feature learning methods, such as Predictability Minimization Layers [14], Restricted Boltzmann Machines (RBMs) [15] and auto-encoders [16].

Here we briefly specify the auto-encoder (AE) framework and its terminology.

Encoder: a deterministic function f_θ maps an input vector $\mathbf{x} \in \mathbb{R}^d$ into hidden representation $\mathbf{y} \in \mathbb{R}^{d'}$: $\mathbf{y} = f_\theta(\mathbf{x}) = \sigma(\mathbf{W}\mathbf{x} + \mathbf{b})$ with parameters $\theta = \{\mathbf{W}, \mathbf{b}\}$, where $\mathbf{W}$ is a $d' \times d$ weight matrix and $\mathbf{b}$ is an offset vector of dimensionality d' .

Decoder: the resulting hidden representation $\mathbf{y}$ is then mapped back to a reconstructed d -dimensional vector $\mathbf{z}$: $\mathbf{z} = g_{\theta'}(\mathbf{y}) = \sigma(\mathbf{W}'\mathbf{y} + \mathbf{b}')$ with $\theta' = \{\mathbf{W}', \mathbf{b}'\}$. The two parameter sets are usually constrained to have tied weights between $\mathbf{W}$ and $\mathbf{W}'$: $\mathbf{W}' = \mathbf{W}^T$.

The parameters are optimized to minimize an appropriate cost function (e.g. measure square error) over the training set.

2.2.1 Denoising Auto-Encoder

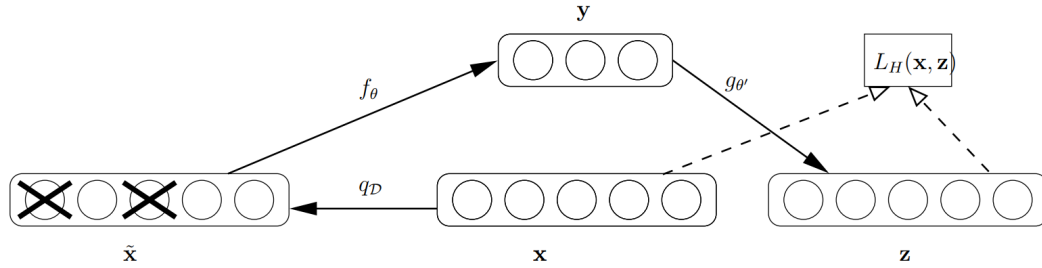


Figure 2.1. The denoising auto-encoder architecture.

In order to make the trained representation robust to partial destruction of the input, Vincent et al. [17] proposed denoising auto-encoders by introducing a corrupted version of the input. As showed in Figure 2.1, during the encoder and decoder process, the representation $\mathbf{y}$ is trained on the corrupted input $\tilde{\mathbf{x}}$ instead of original input $\mathbf{x}$, while the cost function is measured between the reconstructed $\mathbf{z}$ and the uncorrupted input $\mathbf{x}$.

2.2.2 Convolutional Auto-Encoder

To deal with 2D image structure with auto-encoder and reduce redundancy in the parameters brought by global features, convolutional auto-encoder (CAE) is proposed [18]. The weights are shared among all locations in one feature map of a channel and the reconstruction is a linear combination of basic image patches based on the latent code.

For the input x of k -th feature map ($0 < k \leq H$, H is the number of latent feature maps), the representation is computed as $y^k = \sigma(x * W^k + b^k)$. Here σ is an activation function and $*$ denotes the 2D convolution. The bias b^k is broadcasted to the whole map. The reconstruction is obtained with: $z = \sigma(\sum_{k \in H} y^k * \tilde{W}^k + c)$. $\tilde{W}^k$ denotes the flip operation over both dimensions of the weights.

Mean squared error (MSE) between the input x and reconstructed z is used to measure the cost function that is to be minimized. As in the standard neural networks, the backpropagation algorithm is applied to compute the gradient of the cost function with respect to the parameters. A max-pooling layer is used to obtain translation-invariant representation.

2.2.3 Stacked Auto-Encoder

Deep networks can be trained by building several auto-encoders in a layer-wise way [19]. The representation of the n -th layer is used as the input for the next $(n+1)$ -th layer and the $(n+1)$ -th layer is trained after the n -th has been trained. This pair-wise greedy procedure has shown significantly better generalization on a number of tasks [20].

2.3 Recurrent Neural Network

Recurrent Neural Network (RNN) was created in the 1980's but was recently becoming popular from the rapid development of network design and increasing computational power from graphic processing units. It is especially useful to deal with sequential data and has gained great success in a large range of natural language processing related tasks [6, 10, 4, 21, 9]. The most important feature of RNN is that each neuron in RNN is able to use its internal memory to maintain information about the previous input. Besides language, other sequential related tasks have also benefit from it, such as sequence of images, sound, video and so on.

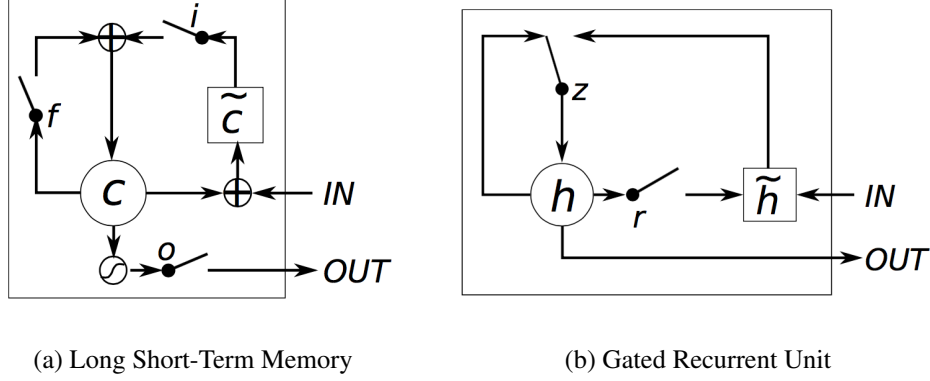


Figure 2.2. Illustration of (a) LSTM and (b) GRU. (a) i , f and o are the input, forget and output gates, respectively. c and $\tilde{c}$ denote the memory cell and the new memory cell content. (b) r and z are the reset and update gates, and h and $\tilde{h}$ are the activation and the candidate activation.

Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) are the two mostly used units, as shown in Figure 2.2. LSTM contain information outside the normal flow of the recurrent network in a gated cell. Similar to data in a computer’s memory, the cell is able to store, write or read information. Through open or close gates, the cell decides to what to store and when to allow reads, writes and erasures. GRU is basically an LSTM without an output gate. At each time step, it fully writes the contents from its memory cell to the larger net.

2.4 Adversarial Training

Adversarial Network was first proposed by Goodfellow et al. in [22] to estimate generative models. It consists of a generative model G that captures the data distribution, and a discriminative model D that estimates the probability that a sample came from the training data rather than G . An adversarial training process corresponds to a two-player minimax game, where the generative model plays the role of counterfeiters, trying to produce fake currency, while the discriminative model acts as the police, trying to detect the counterfeit currency. In particular, the real data samples are labeled as positive class (i.e., 1) while the generated fake samples are labeled as negative class (i.e., 0), and D is trained as a two-class classifier. As a result, D captures the high-level

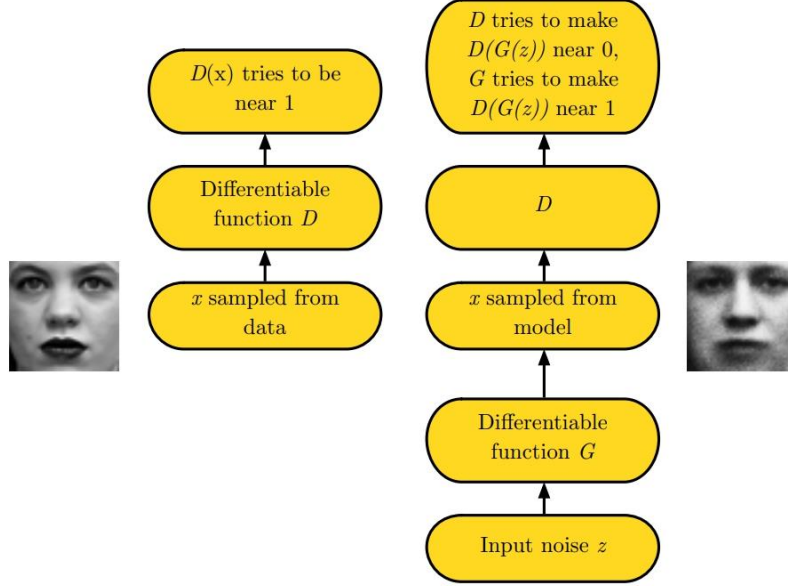


Figure 2.3. An illustration of generative adversarial framework. The generative model G takes as input a noise vector z and generate an image of human face, while the discriminative model (D) first learns the difference between the real face images and the generated images and then optimize G by switching the label of the generated samples from 0 to 1.

difference between the distribution of real and fake samples, which is further used as a guidance of G through back propagation by setting the label of generated samples to 1 (shown in Figure 2.3). Such a two-player minimax game can be formulated as:

$$\min_G \max_D V(D, G) = \mathbb{E}_{\mathbf{x} \sim p_{data}(\mathbf{x})} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_z(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))] \quad (2.2)$$

Using discriminators to optimize the distribution of data by constraining its high-level feature is not only novel but also effective. Therefore, adversarial network is widely used and has been proved being powerful for many fields, e.g., image translations [23, 24, 11], image inpainting [25], image caption [26] and storytelling [1].

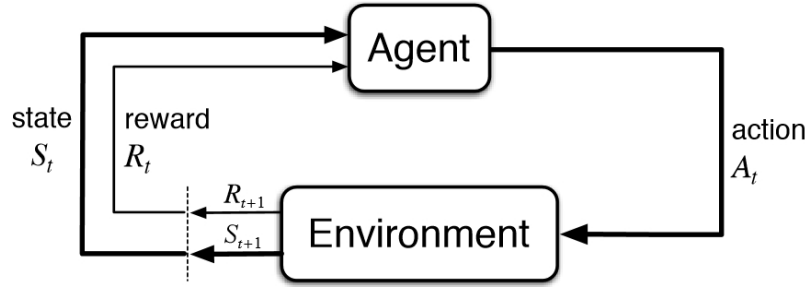


Figure 2.4. An illustration of reinforcement learning system.

2.5 Reinforcement Learning

Reinforcement Learning (RL) [27, 28, 29] is a branch of machine learning in which an agent learns from interacting with an environment. RL differs from standard supervised learning in that correct input/output pairs need not be presented, and sub-optimal actions need not be explicitly corrected. Instead the focus is on performance, which involves finding a balance between exploration (of uncharted territory) and exploitation (of current knowledge). An RL framework allows an agent to learn from trial and error. The RL agent receives a reward by acting in the environment and its goal is learning to select the actions that maximize the expected cumulative reward over time. In other words, the agent, by observing the results of those actions that it is taking in the environment, tries to learn an optimal sequence of actions to execute in order to reach its goal.

An RL system is composed of agent, environment, state, action and reward (as shown in Figure 2.4). With rising interest in research works on deep learning in the middle of the 2000s decade, the promise to use neural networks as function approximator both for the state value function and the action-value function in visual based RL tasks came back. And such a mechanism benefits various of fields such as fine-grained image recognition [30], object detection [31] and image caption [32].

CHAPTER 3

EVENT SUMMARIZATION WITH IMAGES

3.1 Introduction

With the popularization of social networks and image-sharing communities such as Facebook, Twitter, Flickr, Pinterest, and Instagram, images are becoming a more common way to express and share our daily life with friends. In addition to their informativeness and visual appeal, images play an important role as an online communication tool. For example, a user can post several images to let his/her followers know about a trip taken to Japan in summer with a description written in Chinese. As the saying goes, *A picture is worth a thousand words*, and his/her friends can quickly grasp the user's experience even without reading the description carefully, and can save time spent in social network communities, which continues to increase year after year. Moreover, even non-Chinese speakers can understand a Chinese user's updates only by looking at the posted images. As seen in the example above, image-based communication has advantages in terms of its efficiency (less time necessary to understand the content), perspicuity (able to view more content at the same time), and language-independence (comprehensible for speakers of any language), although it cannot be correctly understood unless appropriate images are used for the communication. Thus, the core technology that enables efficient image-based communication is to automatically gen-

erate a *cognition-aware summary of images* that can be accurately recognized by other users.

In this work, we propose a way of tackling the problem of generating a cognition-aware summary of images for a given event. Cognition-aware summaries of images for an event are defined as those with high *perceptual quality* for the event, which is defined as the accuracy and quickness to recognize an event by the summaries. For the example of “traveling to Japan in summer”, we expect as output images that might include “shinkansen bullet train”, “old Japanese temples”, and “the user wearing a kimono”. In contrast, an image summary is not considered to be cognition-aware if it contains images that cannot be recognized as “traveling to Japan in summer”, for example, “cars in Japan” and “Japanese ships” (since their appearance is not much different from those in other countries). As we discussed earlier, a cognition-aware summary of images can enable language-independent, efficient communication in a social network and has many applications including event-driven image thumbnails for a personal image collection and a communication tool in multilingual communities.

Two main differences exist between our research and existing work on image summarization. First of all, the purpose of our work is to improve the perceptual quality of an image set, which differs from that of existing work, which is to extract an informative overview of an image collection [33][34]. It follows that our approach also differs from the approach used in image summarization. For example, Sinha et al. [35] addressed the problem of summarizing personal images from life events and aimed to best represent an image collection with its small representative subset. However, such an image set does not necessarily have high perceptual quality, as their approach does not consider other similar events in which a resulting image set can be misrecognized. Suppose that a user wants to summarize images taken during his/her trip to the Kansai area in Japan. When the summary consists of sights in Kyoto, Japanese food in Osaka, a shinkansen train in Kobe, and women wearing kimono in Nara, one can easily misrecognize the summary as “trip to Kyoto” since images of Japanese food, shinkansen trains, and women in kimono would also be common in images of Kyoto. While the summary does not include any specific images of the Kansai area, it will be misrecognized as a “trip to Japan” instead. From the viewpoint of perceptual quality, the image set should be generated by taking into account the event’s neighbor event (e.g., “trip to Kyoto”, “trip to Japan”) and minimizing possible misrecognitions (e.g., including images of landmarks in several cities in Kansai). Therefore, we analyze the relationship between events and summarize possible neighbor events that can easily cause

misrecognition. Then we propose a method to prevent users from misrecognizing an image set as neighbor events.

Three types of neighbor events are defined for those events that might cause misrecognition: sub-events, super-events and sibling-events. We study the reasons for these misrecognitions and then put forth three criteria to minimize the misrecognitions, namely, sub-event coverage, super-event coverage, and difference from sibling events. A greedy algorithm is applied to integration of three criteria for generating approximately optimal image set with high perceptual quality.

Experiments were conducted using a considerable number of images from Flickr, and the experimental results showed that our proposed method was able to achieve high perceptual quality in comparison with a baseline method, although the quality highly depends on the queries used and size of the generated image sets. The results also indicated that image summarization technologies are not always effective for generating cognition-aware summarization of images.

Three contributions of this work are briefly described:

1. We raise the problem of cognition-aware summarization of images for a given event from the standpoint of viewers who expect to find important points and events from images, which is different from the problem addressed in existing research on image summarization (Sections 3.1 and 3.2),
2. We propose a way to solve the problem by using the method of generating an image set with high perceptual quality by minimizing possible misrecognitions, namely, sub-event, super-event and sibling-event misrecognition (Sections 3.3 and 3.4), and
3. We explain the effectiveness of implementing our approach in experiments using a large number of images from Flickr. The results were evaluated and compared with a baseline method based on an existing image summarization method (Section 3.6).

We present a conclusion in Section 3.7.

3.2 Related Work

This research concerns two main problems: event representation and image summarization. We review here other studies related to these two problems and point out the

differences between them and our work.

3.2.1 Event Representation

Event representation has been explored in several different study fields for years. Language and cognition have opened up the field of how the mind deals with the experience of events [36]. As in computer science, there are studies focusing on how to detect and represent events in natural language, and with the evolution of social networking systems (SNSs), some studies have been done on processing social events[37][38]. In the multimedia domain, event detection has been widely researched with both videos and images. Different approaches have been proposed to understand the structure of complex events in videos [39][40].

In recent years, a lot of research has been done on the prediction of events from images. Chen and Roy [33] proposed an approach to detect Flickr images depicting events. Brenner and Izquierdo [41] incorporated different features to detect social events from collaboratively annotated image collections. These studies solve the problem of detecting events and clustering images while not considering the characteristics of events and the relationship between events. In this work, we focus more on how to select cognition-aware images to represent an event by paying attention to the concerned events' relations.

3.2.2 Image Summarization

Most research on image summarization has focused on personal image collections. Platt et al. [42] presented an overview of a user's image collections generated by an image clustering algorithm that considered time and color. Other features such as time, location [43], social context features (such as tags and comments) [44], and blog posts[45] were utilized to help with image summarization.

Unsupervised approaches were proposed in [46] for event clustering based on time and image content. An event-clustering algorithm was developed to automatically segment images into events and sub-events for albuming based on date/time information and color content of the images [47]. PageRank was employed to mine the most informative information from images of the same event in [48].

The metrics of a good image summarization differ. Sinha and Jain [44] proposed generating a summary on the basis of three properties: relevance, diversity, and coverage. While Wen and Lin [49] focused on two metrics: representativeness and diversity.

The goal of this work is different from the above studies on image summarization. We focus more on the accurate recognition of a specific event from our generated image set, which is emphasized on finding cognition-aware images.

3.3 Preliminaries

In this section, we will first define the events that we focused on and then explain the problem we addressed in this research.

3.3.1 Events

A common definition of an event is “a segment of time at a given location that is conceived by an observer to have a beginning and an end” [50]. Many studies have focused on specific events such as current news items, sports events or earthquakes [33][51][52].

In this research, we are tackling real life events, which are normal events related to our everyday life, for example, a “football game”. We refer to some existing research about general events [53], and define events in this work as:

Definition 1 (Event). *An event involves human activities and can be specified with time and location.*

An event can be denoted with certain terms, e.g., “travel Japan” and “hiking summer”, where the terms “travel” and “hiking” represent activities, while “Japan” and “summer” specify the location and time of each event.

Note that many terms can imply an activity depicted in images, including terms in verb or noun form, since everything in the image can reflect what an image taker is doing while taking that image. For example, the term “lavender” indicates a plant and cannot be regarded as an activity in a common sense. However, if an image depicts lavender, it means the image is taken when a user is viewing the lavender. As a result, the term “lavender” also indicates the activity of “viewing lavender” by taking the activity of the image taker into consideration. Therefore, in this research, the terms describing the activities involved in an event are not restricted to verbs when the event is shown with images.

3.3.2 Problem Definition

The problem we tackle in this work can be defined as:

Definition 2 (Event Representation Problem). *Given an event $e \in E$, a collection of images P , and the size of an output image set n , return image summarization with an image set $S_e \subset P_e$ of size n that maximizes perceptual quality,*

where E represents all possible events, and P_e refers to images that are related to event e among the whole image dataset P . Perceptual quality is defined as:

Definition 3 (Perceptual Quality). *Perceptual quality of an image set for an event is the metric that measures the accuracy and quickness of the image set being recognized as the event.*

Higher perceptual quality makes an image set easier to recognize as a certain event.

Image collection P is a large set of images that were taken by different users, while P_e can be images of event e taken by many users as well as one user. Users can decide an appropriate size of the output image set depending on their application, such as thumbnails and cover images for a personal image library or SNS posts. As a matter of fact, current SNSs also limit the number of images a user can upload for a new update. For example, Twitter allows four images for each tweet, and Weibo (Chinese twitter) permits up to nine images for each tweet. Although Facebook can accept more images for one update, it can display around five images as a thumbnail for each update. Therefore, a small image set is preferred, especially within the size of ten or fewer images. Thus, we aim to produce a cognition-aware image set representing events with image set size of fewer than nine images, which is often used as an upper size limit when combining images using image editing softwares.

For each event $e \in E$, there are many neighbor events that may cause misrecognitions.

Definition 4 (Neighbor Events). *Neighbor events of an event are events that are similar to the event and can easily cause misrecognitions.*

The definition of misrecognition is as follows:

Definition 5 (Misrecognition). *Misrecognition happens when something (such as an event) is recognized as something else.*

In order to find an image set with high perceptual quality, we should avoid neighbor events that will cause misrecognition.

3.4 Approach

In this section, we introduce our approach to generate an image set with high perceptual quality.

3.4.1 Overview

As we mentioned in Section 3.1, our goal is to find an image set that represents an event with high perceptual quality so that users can recognize that exact event after viewing the image set. We introduce an assumption that:

Assumption 1. *An image set that is not likely to be misrecognized as another event is likely to be recognized accurately.*

With this assumption, we propose a method to minimize misrecognitions from neighbor events that can easily cause misunderstandings and to accordingly achieve high perceptual quality. Sub-event coverage, super-event coverage, and the difference from sibling events are used as quantized criteria, and they are proposed to prevent three misrecognitions.

A combination of the three criteria is used to make the objective function, and a greedy algorithm is applied to generate an approximate optimal image set.

We also propose to generate neighbor events by considering the relationship between events. Visual features, including global and local features, and social features such as tags attached to images are employed in our approach.

3.4.2 Misrecognition Situations

There are three types of misrecognitions that can occur when a user looks at an image set and tries to determine which event it is, and they are mainly caused by three types of corresponding neighbor events:

1. The first kind is sub-event misrecognition, which means users recognize an image set of an event as a sub-event of the original event.

Definition 6 (Sub-Events). *Event A is a sub-event of event B if A can only represent part of B.*

For example, the event “travel” consists of several sub-events such as “transport” (actually it is “travel transport”, but we removed “travel” for simplification), “shopping”, and “sightseeing”. Given the event “travel” to be recognized, an image set that includes only images of “transport” will cover one single sub-event of “travel”. This image set has a high possibility of being recognized as “travel transport”. Intuitively, we can avoid sub-event misrecognitions by generating an image set that covers as many sub-events as possible.

2. The second situation is called super-event misrecognition, which means an image set is thought of a super-event of the original event.

Definition 7 (Super-Events). *Event B is a super-event of event A if only part of B can be represented by A.*

Super-events correspond to sub-events. When an image set consists of only images that are common in its super-event, super-event misrecognition will occur. For instance, super-events of “travel in Kyoto” are “Kyoto” (all the events that can happen in Kyoto) and “travel”. Moreover, images of “sakura”, “temple”, and “food” are often taken for the event “travel in Kyoto”. However, they can also represent the event of “Kyoto”, which refers to all the activities that may happen in Kyoto. If an image set only includes images of Kyoto but does not include any images portraying travel, people prefer to treat it as its super-event “Kyoto” when they cannot identify more specific content. A simple solution to avoid this type of misrecognition is to cover all possible super-events within an image set.

3. Sibling-event misrecognition is the last type of misrecognition, and it indicates the case where a user misrecognizes an image set of an event as its sibling-event.

Definition 8 (Sibling-Events). *Event A is a sibling event of event B if A and B can both represent different parts of the same super-event.*

For example, an image set for “conference party” includes images of “people communicating”, “dishes and desserts”, and “proposing a toast”, and all of them can be recognized as “conference party”. However, without labels indicating “conference party”, it will be difficult for a user to tell what kind of party the set is about. Users may take it to be a “birthday party” if an image includes cakes. “Birthday party” and “conference party” have the super-event “party”, so each

of them presents a specific part of the event “party”. We can avoid this kind of misrecognitions by using an image that does not imply a specific kind of sibling-events, such as an image of people gathering rather than an image of a cake in this example.

3.4.3 Sub-Event Coverage

In order to avoid the first type of misrecognition—sub-event misrecognition—we propose the following assumption:

An image set that covers only a single or a few sub-events may cause sub-event misrecognitions.

Under this assumption, an ideal image set should cover as many sub-events as possible. By using the example we used in Section 3.4.2, an image set that contains “transport”, “food”, and “landscape” would be better to represent the event “travel” than one that only contains “transport”. Thus, the sub-event coverage is used to evaluate how likely an image set can prevent sub-event misrecognitions.

Figure 3.1 also shows an example of sub-event coverage. Image set (B) covers only images of traveling in Kyoto, which is just one part of travel in Japan, and users might easily perceive that the images represent “Kyoto travel in spring” rather than “Japan travel in spring”, and this is a sub-event misrecognition. In contrast, if an image set, for example (A), covers images of travel in different parts of Japan such as Kyoto, Tokyo and Mt. Fuji, users will not be misled about the specific place, and the correct position “Japan” can be easily ascertained.

The sub-event coverage can be measured by borrowing an idea in search result diversification, which aims to retrieve search results that cover as many topics as possible in response to a given query [54]. The approach used in search result diversification is to estimate the probability that all the topics will be covered with at least one search result, and to find a set of search results that maximizes this probability. Therefore, as with search result diversification, we estimate the probability that all the sub-events will be covered with at least one image and try to find a set of images that maximizes this probability. Thus, the sub-event coverage $\text{SubCov}(S, e)$ is defined as follows:

$$\text{SubCov}(S, e) = \sum_{v \in \text{Sub}(e)} P(v|e) \left(1 - \prod_{s \in S} P(c = 0|s, v)\right), \quad (3.1)$$

where e is a given event, S is an image set, $\text{Sub}(e)$ refers to sub-events of event e , $P(v|e)$ is the probability that event e contains sub-event v as well, and $P(c = 0|s, v)$ is the



Figure 3.1. Example of Sub-Event Coverage: image set with only images of “Kyoto” (B) will indicate “Kyoto travel spring Japan”, which is a sub-event of “Japan spring travel”, while an image set that covers different parts of Japan (A) will not have this problem.

probability that image s does not present sub-event v with c (binary value) to indicate whether it is presented (value 1) or not (value 0). We assume a unique distribution for $P(v|e)$ due to the lack of prior knowledge for this probability, i.e.,

$$P(v|e) = \frac{1}{|\text{Sub}(e)|}. \quad (3.2)$$

An intuitive interpretation of this formula is that $\text{SubCov}(S, e)$ becomes high if at least one of the images in an image set S has high probability $P(c = 1|s, v)$ for all the sub-events of e .

Below, we discuss a method of estimating the probability $P(c = 1|s, v)$, which is a complement of $P(c = 0|s, v)$, $P(c = 1|s, v) = 1 - P(c = 0|s, v)$. A basic assumption here is that image s is likely to cover sub-event v if s is similar to images that were taken in sub-event v . We used k -nearest neighbor distance $k\text{-NND}(s, v)$, which is the average distance of k -nearest neighbor images of image s in image set P_v , to measure the similarity between image s and images of event v [55]. In addition to its simplicity, the computation of the k -nearest neighbor distance is efficient, since k -nearest neighbor

search has been extensively studied in the literature. We obtain the following formula by taking the inverted distance of k-NND(s, v) with an exponential function:

$$P(c = 1|s, v) = \exp(-\lambda \cdot \text{k-NND}(s, v)), \quad (3.3)$$

where λ is used to control the shape of this distribution.

In summary, sub-event coverage $\text{SubCov}(S, e)$ measures how likely an image set can prevent sub-event misrecognitions. With a higher $\text{SubCov}(S, e)$ value, image set S is able to cover more sub-events of event e . An image set with high sub-event coverage is expected to avoid sub-event misrecognitions and consequently achieve high perceptual quality.

3.4.4 Super-Event Coverage

Super-event misrecognition can happen when images in an image set are only related to part of the original event’s super-events and cannot cover all of them. This observation led to an assumption that:

An image set that covers only one or a few super-events may cause super-event misrecognitions.

An image set S that can prevent this type of misrecognition covers all the super-events $\text{Sup}(e)$ of event e with at least one image in the image set S . In the example of image set (A) in Figure 3.2, when “travel” and “Kyoto” are both emphasized in the image set, it will be easier to recognize both events and form the event “travel in Kyoto”. Thus, we apply the algorithm of sub-event coverage in Equation 3.1 to each super-event and compute the super-event coverage of original event $\text{SupCov}(S, e)$, as described in Equation (3.4).

$$\text{SupCov}(S, e) = \sum_{u \in \text{Sup}(e)} P(u|e) \left(1 - \prod_{s \in S} P(c = 0|s, u)\right). \quad (3.4)$$

Only if an image set covers all super-events will it achieve high super-event coverage. For example, an image set with high super-event coverage for the event “travel in Kyoto” would be one that has high sub-event coverage for both the events “travel” and “Kyoto”. Higher super-event coverage guarantees that all super-events are covered by an image set.

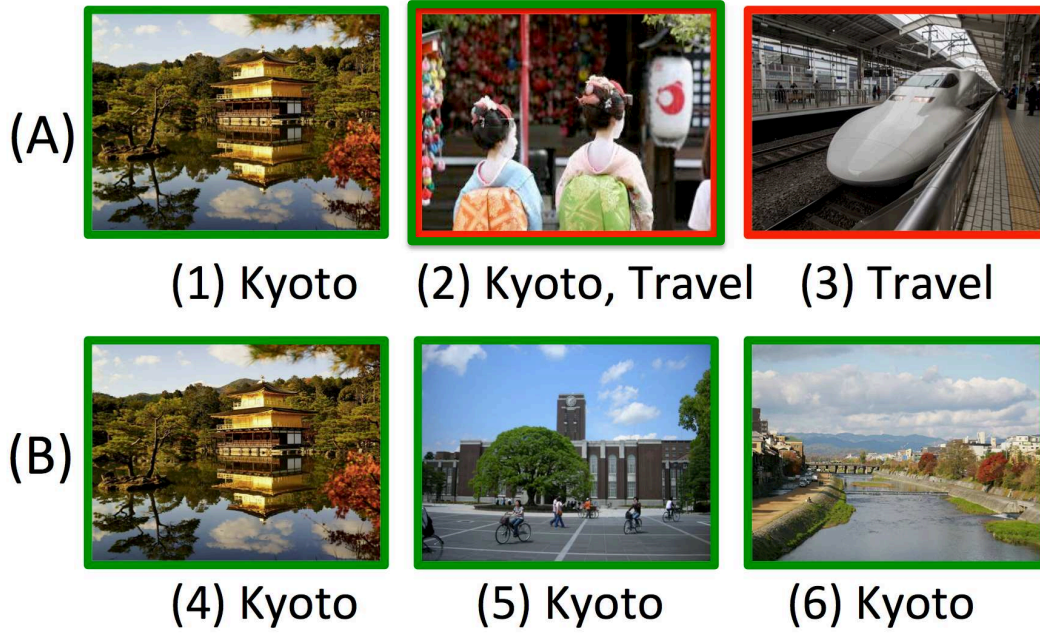


Figure 3.2. Example of Super-Event Coverage: image set with only images of “Kyoto” (B) will indicate “Kyoto”, which is a super-event of “Kyoto travel”, while an image set that covers both “Kyoto” and “travel” (A) will not have this problem.

3.4.5 Difference from Sibling-Event

To avoid sibling-event misrecognition, we should not use images that present only a few sibling-events.

An image similar to just a few sibling-events can cause sibling-event misrecognitions.

Thus, images that are similar to all sibling-events or not similar to any sibling-event under one super-event are preferable in order to avoid sibling-event misrecognition. With images that are similar to all sibling-events under one super-event, our consideration is that all the sibling-events should contain common features of the same super-event. For example, to find images of “travel Japan”, an image that is common in all sub-events of “travel” is generally sufficient to avoid sibling-event misrecognition. However, since super-event coverage guarantees that all the super-events are covered in the output image set, there is no necessity to use the idea again in preventing sibling-event misrecognition. To this point, we propose a difference from sibling-events to

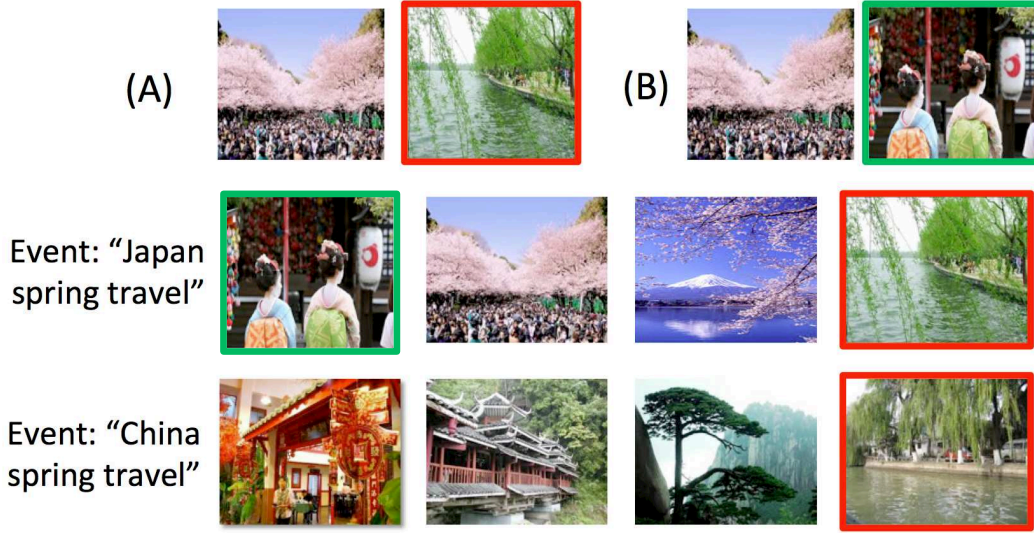


Figure 3.3. Example of difference from sibling-events. Images of willow trees are common in both Japan and China in spring, so an image set that includes an image of a willow tree (A) is not as good as one that includes an image of women in kimono (B), since an image of women in kimono is unique to Japan and will not cause misrecognition.

avoid misrecognition from sibling-events. The formula is as follows:

$$\text{SibDif}(S, e) = 1 - \prod_{s \in S} \max_{v \in \text{Sib}(e)} P(c = 1 | s, v) \quad (3.5)$$

where $\text{Sib}(e)$ are all sibling-events of e under all its super-events. The distribution for $P(c = 1 | s, v)$ is assumed to be equal, just as in Section 3.4.3, which measures the probability that image s covers event v . For each image in an image set, we hope that the maximum possibility of covering sibling-events is as small as possible. Note that this criterion is used to complement sub-event coverage and super-event coverage.

As a result, images that are different from all sibling-events will rank higher. In the example in Figure 3.3, willow trees are quite common in both "spring in China" and "spring in Japan", while they are not as common in other places; thus, images of willow trees are often taken in spring in these two places. If we use an image of a willow tree to represent the event "Japan spring travel", users cannot decide where it is and may think it is in China. Instead, images of women wearing kimono are most likely to be taken in Japan, and it is much less common for other places to have similar images. Thus, it will achieve a higher score than an image of willow trees in consideration of

sibling-events of “Japan spring travel”.

3.4.6 Image Set Generation

By combining three criteria to avoid the three types of misrecognition, we generate one objective function to measure to what extent an image set can minimize misrecognitions. Sub-event coverage, super-event coverage, and the difference from sibling-events are combined in order to maximize the objective function $f(S, e)$.

$$f(S, e) = \alpha \text{SubCov}(S, e) + \beta \text{SupCov}(S, e) + \gamma \text{SibDif}(S, e), \quad (3.6)$$

where α , β , and γ are parameters that determine which criteria should be emphasized.

Our objective can now be reformulated as a problem of finding an image set of size n for a given event e that maximizes the objective function $f(S, e)$. Unfortunately, finding an optimal image set is an NP-hard (Non-deterministic Polynomial-time hard) problem.

Lemma 1. $f(S, e)$ is NP-hard.

When images can belong to multiple image sets, there may not exist a single ordering of image sets such that the objective function of $f(S, e)$ is maximized for all possible S . The reason is that a set of images optimal for $f(S', e)$, where $|S'| = n - 1$, need not be a subset of the optimal value of $f(S, e)$, where $|S| = n$.

Because the set function $f(S, e)$ is monotonic and submodular (see the Appendix), we can apply the greedy algorithm and guarantee that the result returns a $(1 - 1/e)$ -approximation of the maximum[56], which often gives a good approximation to the optimum. We start with an empty image set S and iteratively add an image $p \in P_e$ to S that maximizes $f(S \cup \{p\}, e)$ until the size of the image set $|S|$ reaches n . The greedy algorithm is described in Algorithm 1.

3.5 Neighbor Event Generation

To find proper neighbor events that can easily cause misrecognition, we utilize images of events and their surrounding social information such as tags.

Algorithm 1 Greedy Algorithm for Image Set Generation

Require: User’s query, event: e ;
A collection of images: P ;
Size of final image set: $n \in \mathbb{N}$;
Ensure: An image set of size n : $S, |S| = n$;

- 1: Generate e related image collection P_e
- 2: $S = \{\}$
- 3: **for** each $i \in [1, n]$ **do**
- 4: $p^* = \operatorname{argmax}_{p \in P_e} f(S \cup \{p\}, e)$;
- 5: $S = S \cup \{p^*\}$;
- 6: $P_e = P_e - \{p^*\}$;
- 7: **end for**
- 8: **return** S

3.5.1 Generating Sub-Events

According to the definition, sub-events are events that can present part of an event. We need to find ones that can easily cause misrecognition. Our strategy of generating sub-events is to add a keyword to the keyword set of the original event and to let the added keyword specify a certain part of that event. As a result, our objective becomes finding keywords to be added and in particular, keywords that can be easily confused with the target event e . Added keywords are selected from tags of event T_e if they comply with two criteria of selected sub-events: they are relevant (representative and visually similar) and irredundant.

A sub-event needs to be representative to event e because representative content is likely to remind viewers of event e . For example, “travel Kyoto temple” is a representative sub-event of the event “travel Kyoto”, and images of “travel Kyoto temple” have a high possibility to be regarded as “travel Kyoto”. Term frequency-inverted document frequency (TF-IDF) is used to determine the importance of representative sub-events. We first compute TF (in P_e)-IDF of all event-related tags T_e [57]. A tag is denoted by t , and $\text{tf-idf}(t, e)$ represents its TF-IDF value.

Images of a sub-event that could cause misrecognition are usually visually similar to images of the event. We calculate the visual similarity between e and each sub-event, which is defined by the image similarity of events e and t , $\text{VisualSim}(t, e)$. Visual similarity between two events is obtained by measuring the Euclidean Distance between

their visual features [58]. Suppose we have two events e_1 and e_2 : images of them are denoted by P_{e_1} and P_{e_2} . Visual features of each tag are obtained by taking the average visual features of all images in P_{e_1} and P_{e_2} . Visual features of each tag are also obtained from global visual features such as color (RGB and HSV), and local features with a 1000-D bag of visual words[59].

The relevance of a tag t to event e is the harmonic mean of its TF-IDF and visual similarity value:

$$\text{SubRel}(t, e) = \frac{2\text{tf-idf}(t, e)\text{VisualSim}(t, e)}{\text{tf-idf}(t, e) + \text{VisualSim}(t, e)} \quad (3.7)$$

To ensure that all selected sub-events are efficient and are able to cover many different aspects of the event, we need to reduce the redundancy of resulting sub-events. Here, redundant sub-events refer to ones that describe almost the same content. For instance, images of “spring Japan” and “spring Nihon” are basically the same. We use context similarity between tags as a diversity metric, since tags that often appear together with the same vector of tags are supposed to present the same content[60]. Context similarity, $\text{ContextSim}(t_1, t_2)$, measures whether two tags are similar by considering their neighbor tags. For example, “cloud” and “sky” are often tagged with the same set of tags in one image, such as “blue, water, tree”, and are therefore very similar based on the tag context.

Then maximal marginal relevance (MMR) [13] is applied to find tags that balance relevance with e and diversity from selected sub-event tag set $W_e \subset T_e$, as shown in the following formula:

$$\text{MMRSub}(t, e) = \arg\max_{t_i \in T_e \setminus W_e} \left[\theta \text{SubRel}(t_i, e) - (1 - \theta) \max_{t_j \in W_e} \text{ContextSim}(t_i, t_j) \right] \quad (3.8)$$

After finding several top sub-event tags with MMR, we generate sub-event v , whose keyword set is denoted as K_v , by adding each sub-event tag to the original event’s keyword set K_e , i.e.,

$$\text{Sub}(e) = \{v | K_v = K_e \cup \{w\} \wedge w \in W_e\} \quad (3.9)$$

Thus, if “cherry” and “Tokyo” are the top tags generated by our method of finding sub-events of “travel Japan”, corresponding sub-events will be “cherry travel Japan” and “Tokyo travel Japan”.

3.5.2 Generating Super-Events

In accordance with the idea of obtaining sub-events by adding one keyword to specify a certain part of an event, super-events are generated by removing each keyword from the original event's keyword set K_e , since subsets with a smaller number of keywords usually represent a more general concept than the original keyword set. Thus,

$$\text{Sup}(e) = \{u | K_u \subset K_e \wedge |K_u| = |K_e| - 1\} \quad (3.10)$$

where K_u is a keyword set of super-event u . For example, super-events of “travel Japan” are “travel” and “Japan”.

3.5.3 Generating Sibling-Events

Sibling-events present different parts of an event's super-event, and our objective is to find those that are likely to cause misrecognition. We produce sibling-events by specifying an event's super-events with one more word. Similar to the case of sub-event generation, the added word should satisfy the criteria of relevance (contextually and visually similar) and irredundancy. For each super-event u of an event e , the absent word k is one that generalizes the event, i.e.,

$$k \in K_e - K_u, (u \in \text{Sup}(e)). \quad (3.11)$$

The added word should be contextually similar to the absent word in the context of a super-event, because words with a similar context in images are often related to similar contents in the images and will easily cause misrecognition. For example, “Kyoto” is similar to “Nara” in the context of the event “travel”, since they are both related to the contents of “temple” and “traditional”.

In addition, the added word should have a short distance with the absent word in regard to visual features since similar visual features can more easily cause misrecognition.

As a result, context similarity and visual similarity between words are combined to compute the relevance, and for this combination, we use their harmonic mean:

$$\text{SibRel}(t, k) = \frac{2\text{ContextSim}(t, k)\text{VisualSim}(t, k)}{\text{ContextSim}(t, k) + \text{VisualSim}(t, k)} \quad (3.12)$$

With only relevance as the criterion to generate sibling-events, we find that there is some reduplication between sub-events and sibling-events. For example, “travel

Japan Tokyo” is one of the top sub-events of “travel Japan” because many people who go to Japan for travel purposes will visit Tokyo (high TF value), and Tokyo travel is highly related to Japan travel, while it is not frequent in other events (high TF and IDF). However, we can understand that “travel Japan Tokyo” is almost the same event as “travel Tokyo”, which is highly possible as a top sibling-event of “travel Japan” when relevance is the only criterion. In order to exclude sibling-events such as “travel Tokyo”, we utilize the Jaccard distance to measure the dissimilarity of new added word t (“Tokyo”) and abstract word k (“Japan”) in super-event u (“travel”) to ensure that sibling-events that are similar to sub-events are excluded. The Jaccard index is used to measure the concurrence of two words in an event by comparing the similarity and diversity of images that contain each word as a tag. The tag Jaccard distance is the complementary to the Jaccard index of tags:

$$\text{TagJacDist}(t, k, u) = 1 - \frac{|P_{t+u} \cap P_{k+u}|}{|P_{t+u} \cup P_{k+u}|}. \quad (3.13)$$

A longer distance means two tags are not often tagged together in many images of an event. Thus, they are better as sibling-events than those that are often tagged together, under the same condition that they are contextual and visually similar to each other.

Thus, the new relevance computation comes to:

$$\begin{aligned} \text{SibRel}(t, k, u) = \\ \frac{2\text{ContextSim}(t, k)\text{VisualSim}(t+u, k+u)}{\text{ContextSim}(t, k, u) + \text{VisualSim}(t+u, k+u)} \cdot \text{TagJacDist}(t, k, u), \end{aligned} \quad (3.14)$$

by multiplying the original relevance by the Jaccard tag distance. In this case, “travel China” will be ranked higher than “travel Tokyo” when finding sibling-events of “travel Japan” under the super-event “travel”.

Semantic similarity, $\text{SemanticSim}(t_1, t_2)$, is used for diversity because tags with the same meaning should be avoided. We implement it by using path similarity, which computes the semantic relatedness of words by counting the number of nodes along the shortest path between words in the “is-a” hierarchies of Wordnet[61]. Let $W_u \subset T_u$ be the current sibling-event tag set under one of the super-events u . The target function of the MMR algorithm is as follows:

$$\text{MMRSib}(t, u, e) = \operatorname{argmax}_{t_i \in T_u \setminus W_u} \left[\phi \text{SibRel}(t_i, k) - (1 - \phi) \max_{t_j \in W_u} \text{SemanticSim}(t_i, t_j) \right] \quad (3.15)$$

With several top tags w with MMR W_u of each super-event u , we obtain sibling events v of event e under u so that the keyword set of v is the result of adding w to u ’s

keyword set, i.e.

$$\text{Sib}(e, u) = \{v | K_v = K_u \cup \{w\} \wedge w \in W_u\}. \quad (3.16)$$

As we have described, these methods to create neighbor events are based on our requirement to find neighbor events that can easily cause misrecognition. We will show the effectiveness of our method compared with the baseline method (TF-IDF) in the next section.

3.6 Experiments

To evaluate the performance of our approach to avoid misrecognition as well as to evaluate the effectiveness of generating neighbor events, we conducted some experiments by using images crawled from Flickr. We assessed the performance of neighbor event generation and image set generation separately.

The image collection in our experiments consisted of more than 2.9 million images crawled from Flickr, which were used as the entire image collection P .

3.6.1 Generating Neighbor Events

In this experiment, we tested 50 events (indicated in Table 3.1). We picked these events from everyday life events by checking the frequently updated events in Flickr. Furthermore, the selected events are ones that have meaningful neighbor events (especially super-events based on our method). Neighbor events of these events were generated with our proposed method and a simple baseline method. In our method, parameters θ and ϕ , which are used to balance the importance of similarity and diversity in generating sub-events and sibling-events, were set according to our preliminary experiments:

$$\theta = 0.7, \phi = 0.7.$$

As for the baseline, we used TF-IDF to rank tags, and we generated sub-events by adding the top-ranked tags to a target event. Super-events were produced in the same way as in our method, while sibling-events were generated by using sub-events of these super-events with the TF-IDF method.

Table 3.4 lists some examples generated with our method and the baseline method. From this table, we find that TF-IDF always generates duplicate events such as “travel Japan landscape” and “travel spring landscape” for sibling-events. In addition, the

Method	Neighbor Event	Result
Our Method	Sub-Event	‘travel Japan spring park’, ‘travel Japan spring cherry’, ‘travel Japan spring Tokyo’, ‘travel Japan spring tree’, ‘travel Japan spring flower’, ‘travel Japan spring sky’, ‘travel Japan spring Kyoto’ ...
	Super-Event	‘travel Japan’, ‘travel spring’, ‘Japan spring’
	Sibling-Event	‘travel Japan outside’, ‘travel Japan artistic’, ‘travel Japan beautiful’, ‘travel spring outdoor’, ‘travel spring urban’, ‘travel spring traditional’, ‘Japan spring scenic’, ‘Japan spring cute’, ‘Japan spring sunlight’ ...
	Sub-Event	‘travel Japan spring flower’, ‘travel Japan spring landscape’, ‘travel Japan spring sky’, ‘travel Japan spring tree’, ‘travel Japan spring water’, ‘travel Japan spring city’, ‘travel Japan spring light’ ...
TF-IDF	Super-Event	‘travel Japan’, ‘travel spring’, ‘Japan spring’
	Sibling-Event	‘travel Japan landscape’, ‘travel Japan sky’, ‘travel Japan city’, ‘travel spring flower’, ‘travel spring landscape’, ‘Japan spring flower’, ‘Japan spring tree’, ‘Japan spring Tokyo’, ‘Japan spring museum’ ...

Figure 3.4. Neighbor Events of “travel Japan spring”

3. Event Summarization with Images

ID	Event	ID	Event
1	birthday party	26	winter london
2	bungee jump	27	award ceremony
3	car accident	28	bar cocktail
4	company meeting	29	bar concert
5	conference party	30	beach wedding
6	cook dinner	31	festival japan
7	fashion show	32	island holiday
8	football game	33	lavender france
9	graduation ceremony	34	outdoor hotspring
10	ice skating	35	paris shopping
11	playing golf	36	rainbow mountain
12	rock climbing	37	thunder city
13	running on beach	38	travel hokkaido
14	sports game	39	travel london
15	sunset sea	40	yoga outdoor
16	walk dog	41	boat autumn
17	walk travel	42	christmas party
18	wedding ceremony	43	concert night
19	road trip summer	44	flight sunset
20	summer hawaii	45	halloween costume
21	sunrise mountain	46	morning walk
22	swim beach summer	47	spring bike
23	travel canada autumn	48	summer hiking
24	travel japan spring	49	surf summer
25	travel usa winter	50	thanksgiving dinner

Table 3.1. 50 events categorized based on whether the event is time-aware or location-aware

resulting neighbor events are not necessarily ones that satisfy our definition of neighbor events. For example, “travel spring flower” is more appropriate as a sub-event than a sibling-event.

However, neighbor events resulting from our method can meet the needs of neighbor event properties. For instance, “travel spring traditional” is a sibling-event of “travel

Japan spring” under the super-event “travel spring”. We can imagine that travelers who go to Japan in spring often visit some temples or shrines, which are all old buildings and represent traditional things. It is easy to mix up travel in Japan with travel in other traditional places. Let us take “travel Japan spring cherry” as another example. Japan is very famous for its cherry blossoms that only bloom in spring. Everyone who travels to Japan in spring will take images of the cherry blossoms. Hence, it is a very important sub-event of “travel Japan spring”, and our method can create it while the baseline method cannot.

Therefore, neighbor events generated by our method are closer to our expectation and correspond to reality much better. In particular, our proposed method was probably able to generate better sibling-events because we considered the absent term of each super-event. In contrast, the baseline method using TF-IDF generated many duplicate sibling-events probably due to the lack of consideration of the similarity between tags.

3.6.2 Generating Image Sets

After obtaining neighbor events that can easily cause misrecognition, we utilized them in computing image sets using the baseline method and our method separately and in combination. The perceptual quality of the generated image sets was evaluated with a crowdsourcing service.

Baseline

VisualRank (VR) [62] was used as the baseline method to generate image sets. VR is an algorithm that applies PageRank to images, and it can be used to mine the most informative features from images that belong to the same event [48]. The formula of VR is as follows: given n images, VR is recursively defined as

$$\text{VR} = dM^* \times \text{VR} + (1 - d)p, p = \left[\frac{1}{n} \right]_{n \times 1}. \quad (3.17)$$

M^* is the column normalized adjacency matrix M , where $M_{i,j}$ is the similarity between image p_i and p_j , which is computed by the Euclidean distance between visual features of two images. In order to guarantee that the resulting image set does not include the same images, we checked the visual similarity of each candidate image and added images to ensure their similarity Euclidean distance is greater than 0.1.

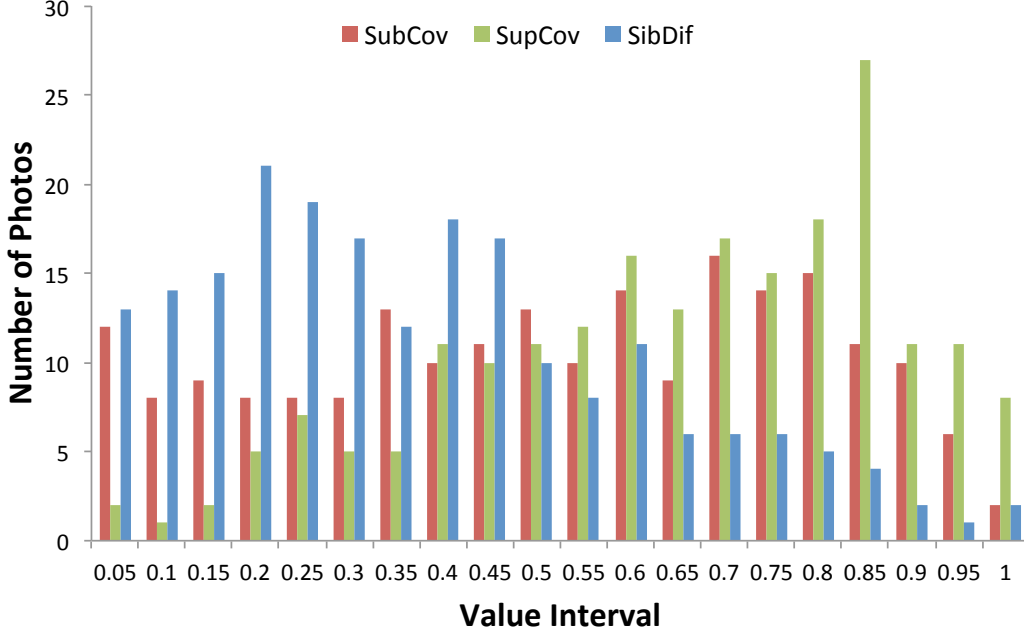


Figure 3.5. Value distribution of three criteria for event “travel Japan spring”.

Setup

We generated image sets with the 50 events listed in Table 3.1 and used five sizes of image sets for each event. Each of these events had more than 100 related images from our image collection. The average number of related images for each event was 307.

We applied five methods, including the baseline VR method and four variations of our proposed method: an objective function with only sub-event coverage (SubCov), only super-event coverage (SupCov), only the difference from sibling-events (SibDif), and an objective function with a combination of three criteria (ALL), which maximizes the value of f in Equation (3.6). After getting the ranking score of event-related images with these five methods, we used the top several images to form an image set.

We conducted preliminary experiments and checked the distribution of three types of scores for each candidate image of the event “travel Japan spring” as displayed in Figure 3.5. The result showed that normalized scores were equally distributed from 0 to 1 for three criteria, which enabled us to give the same parameter to each of the criteria and guarantee their equivalent importance in the target image set. As a result,

three parameters in the objective function were set as follows:

$$\alpha = 1/3, \beta = 1/3, \gamma = 1/3.$$

As mentioned in Section 3.3.2, an image set is usually not larger than nine images. For this reason, we used five sizes of image sets, i.e. $n = 1, 3, 5, 7$, and 9. In total, there were 1250 image set (event, method, size of image set) combinations. The parameter k of the k -nearest neighbor was set to 20, and the parameter λ was set to 25 in this experiment based on our preliminary test.

We used a crowdsourcing service to evaluate the perceptual quality of each image set in two steps: labeling events to generated image sets, and then obtaining the perceptual quality of an image set by comparing the labeled event and the input event. Lancers *, a crowdsourcing service in Japan, was used in our evaluation.

In the first task, five assessors were assigned to each image set, and they were asked to label what they thought the images represent. According to the previous definition, the perceptual quality of an image set is better weighted by the accuracy and quickness of the user’s perception of an event by looking at an image set. In our experiment, we focused on the accuracy component by setting the quickness at a certain level while leaving the evaluation of quickness as an open issue due to practical difficulties. We asked the assessors to label what events they could recognize from an image set in a few seconds, including the time, location, and activity.

The perceptual quality of an image set was measured in the second step by evaluating the agreement between the input event of the image set and the labels the assessors added.

Definition 9 (Estimated Perceptual Quality). *Estimated perceptual quality of an image set S to represent an event e is the average agreement between the event e and events E_S , where E_S denotes events that users can perceive from image set S .*

The agreement between two events was measured on a three-point scale: mismatch (score 0), partial match (score 1), and match (score 2). The assessor marked a label as a match if the label had the same meaning as the event, while the assessor gave a partial match to labels that partially overlapped keywords of the event. For example, “travel Japan” is a partial match, while “spring Japan travel” is a match to the event “travel Japan spring”.

*<http://www.lancers.jp/>

Method: VR
Photo Size: 3



Method: SibDif
Photo Size: 3



Method: ALL
Photo Size: 3



Figure 3.6. Image sets of event “walk dog” generated by different methods: VR (baseline VisualRank), SibDif (difference from sibling-events), and ALL (combination of three criteria).

To guarantee high quality of the assessment, we added a dummy event-label pair every 9 pairs to make up one unit. The assessors gave scores in units and each of them could assign scores for up to 100 units. Scores marked by assessors who made a mistake on the dummy pair were excluded in the final result.

Results

Figure 3.6 shows image sets of the event “walk dog” generated by the VR, SibDif and ALL methods. As we can see, VR gives very similar images but fails to cover all sub-events such as “walk dog leash”. It also lacks differentiation from sibling-events such as “raise dog” and “walk stroll”. SibDif successfully avoids misrecognition from these sibling-events.

Figure 3.7 plots an overall comparison of the results of the five methods. The horizontal axis is the number of images in the image set, and the vertical axis is the average perceptual quality of each size, which is computed by the average score of agreement

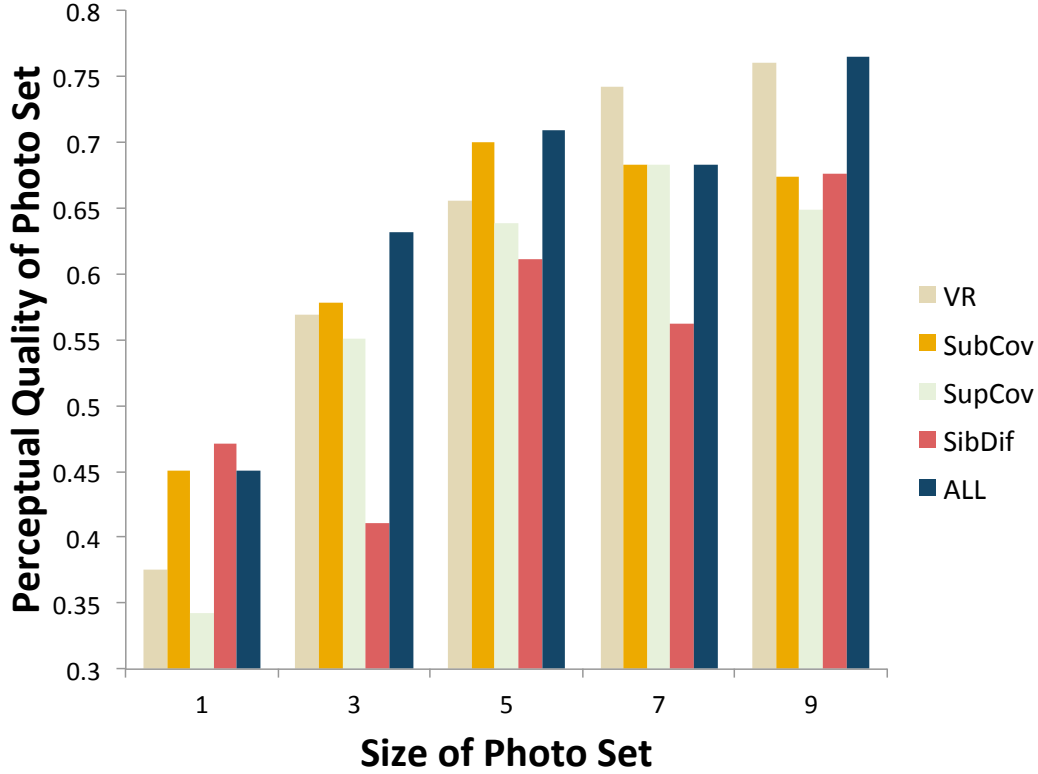


Figure 3.7. Perceptual quality of five methods: we applied each method to 50 events in five different sizes of image sets: 1, 3, 5, 7, and 9.

between target events and labeled events. The overall trend is clear: perceptual quality is gradually improved with more images in an image set. With more than five images, the benefit of adding more images begins to decrease.

We conducted a three-way ANOVA (Analysis of Variance) to test whether there was a difference between the effects of the methods, the sizes of an image set, and the events and their interactions to the perceptual quality of an image set. The ANOVA showed a significant difference in all of them: the type of method ($F(4, 2500) = 6.1$, $p < 0.01$), size of image set ($F(4, 2500) = 51.26$, $p < 0.01$), and events ($F(49, 2500) = 35.5$, $p < 0.01$). Significant interactions were found among all combinations. These values demonstrated that the perceptual quality of an image set varies with different sizes, different methods perform in various ways, and events can affect the performance.

From Figure 3.7, we can also find that three criteria (SubCov, SupCov, and SibDif) show different performances with different image set sizes. The performance of

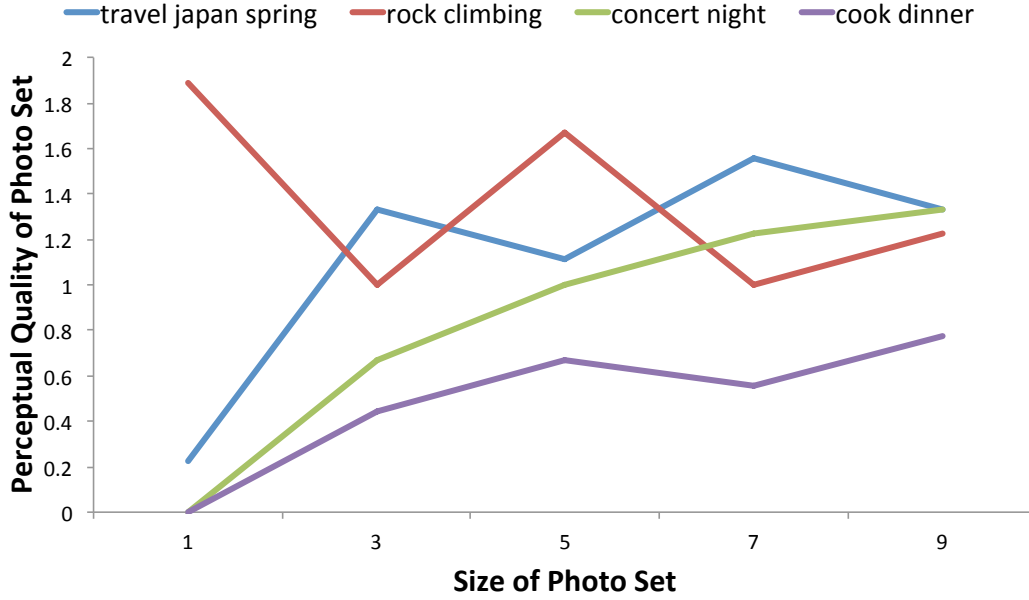


Figure 3.8. Perceptual quality of several events in different sizes.

SubCov increases steadily until size five and stops rising, or even drops after adding more images. This might be due to its coverage of good representative content of an event. With too many images, high sub-event coverage will include images with less important content, which confuses viewers.

SibDif is used as a supplementary criterion to avoid sibling-events. The result indicates that the performance is unstable. Images that portray totally different events will also intuitively achieve a high score with SibDif, which results in atypical images and lower perceptual quality. However, it exhibits best performance with one image. This may be attributed to the fact that it maximizes an event’s difference from sibling-events that can cause misunderstandings, and users can easily figure out the right event without hesitation.

ALL performs best for most image set sizes. It guarantees coverage of the important content of an event and maximizes the difference from similar neighbor events. Compared with the baseline method (VR), it demonstrates better performance with five or fewer images. As discussed in the performance of sub-event coverage, having more images can bring confusing and noisy information that may mislead viewers into perceiving it as another event. To check whether it is true with ALL method, we check four events and their performances of different sizes. As we can see from Figure 3.8,

events that contain a few scenes such as “rock climbing” can be easily recognized in a small image set, while events that are more abstract and contain many different scenes gain better performance with more images. Therefore, more images do not necessarily convey accurate information, so a properly sized image set is preferable.

Method of generating image set	Size of image set	PQ(TF-IDF)	PQ(our method)
VR	1	0.31	0.38
	3	0.51	0.57
	5	0.54	0.66
	7	0.62	0.74
	9	0.61	0.76
SubCov	1	0.31	0.45
	3	0.54	0.58
	5	0.50	0.70
	7	0.53	0.68
	9	0.57	0.67
SupCov	1	0.30	0.34
	3	0.56	0.55
	5	0.58	0.64
	7	0.55	0.68
	9	0.57	0.65
SibDif	1	0.19	0.47
	3	0.30	0.41
	5	0.43	0.61
	7	0.47	0.56
	9	0.53	0.68
ALL	1	0.30	0.45
	3	0.49	0.63
	5	0.55	0.71
	7	0.61	0.68
	9	0.65	0.76

Table 3.2. Comparison of perceptual quality of image sets generated by our proposed method and TF-IDF method.

In addition, we compared the final perceptual quality of image sets for different

neighbor events (generated by our proposed method and the TF-IDF method) to investigate whether the proposed neighbor events contribute to the final performance. The results are given in Table 3.2. Those numbers indicate that our proposed method improved the final performance of generating image sets with higher perceptual quality by providing more suitable neighbor events.

In conclusion, our approach is able to generate image sets of a proper size with higher perceptual quality compared with the baseline method.

3.6.3 Analysis of Events and Perceptual Quality

As indicated from the ANOVA, we can see that events play a role in affecting the performance of perceptual quality. To see the relation between an image set’s perceptual quality and event categories, especially regarding time and location, we classify all 50 events into four types based on whether the event is time-aware or location-aware.

The average perceptual quality of different types of events is shown in Figure 3.9 for five methods. This figure reveals that for all methods, perceptual quality is highest when time and location are not specified; in particular, SubCov reveals the best performance. Time is more easily represented than location with images since the perceptual quality performance of most methods is higher with time-aware events than with location-aware events, except for SibDif. This result conforms to the fact that time is not easily confused because human beings hold almost the same perception about time, such as whether it is night or autumn. However, people see locations in different ways and might mix them up with other similar locations. That also explains why SibDif can better perform with location-aware events.

Moreover, ALL achieves high perceptual quality for most types of events except for location-specified ones. This result demonstrates that our approach fails in presenting location-aware events, although it can generate good cognition-aware image sets for events in which time or location is not specified.

3.6.4 Image Set Size Selection

In our experiment, the size of an image set was given in advance. Now we propose a method to automatically determine the appropriate size of an image set for a given event. The size was compared with sizes we set in the previous experiment by balancing perceptual quality and small sizes.

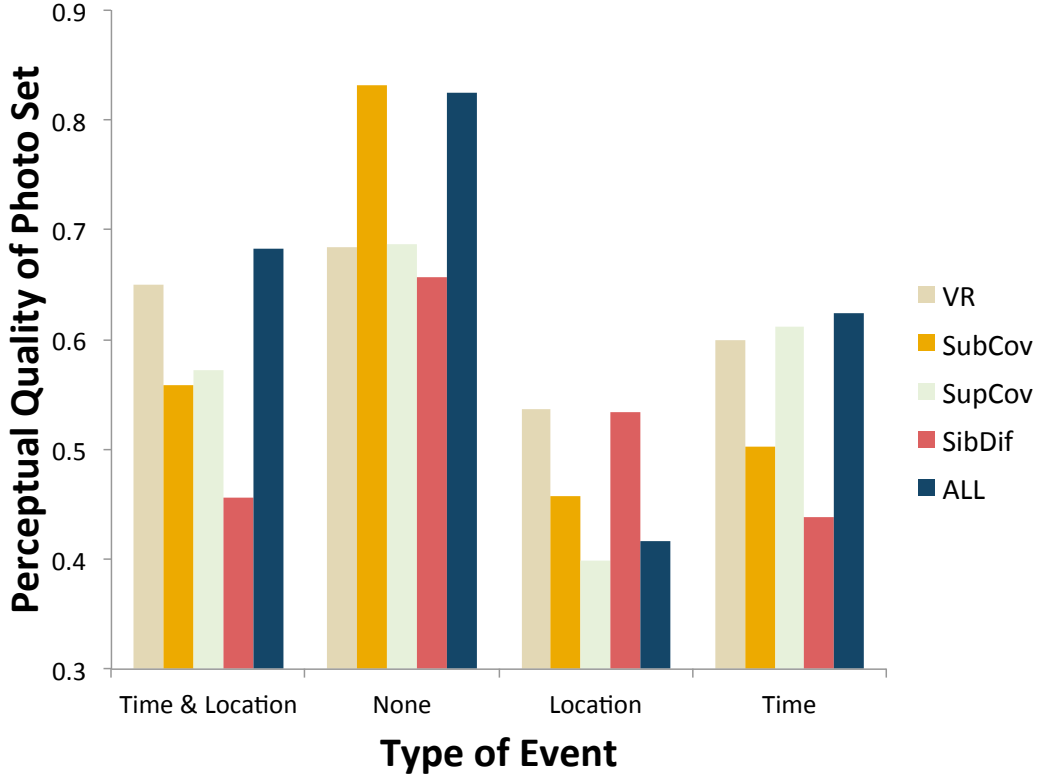


Figure 3.9. Perceptual quality of different types of events and methods. Here, “None” corresponds to non-time specified or non-location specified, “time” to time-specified, “location” to location-specified, and “time & location” to time specified and location specified.

We can see from the comparison of perceptual quality at different sizes that the benefit of adding another image begins to decrease after a certain size, which means we can determine the appropriate size by finding the point where perceptual quality and smallness of the image set size is balanced. Our approach is as follows; in every step of the greedy algorithm shown in Algorithm 1, we check the objective function value and compare it with previous values. If the benefit of the added image is smaller than our required threshold σ , the iteration will end, and previous images will be returned as a result.

Let S_{k-1} and S_k be the result of the $(k-1)$ -th and k -th iteration during the process of the greedy algorithm. When another image p is added to image set S_k in the $(k+1)$ -th

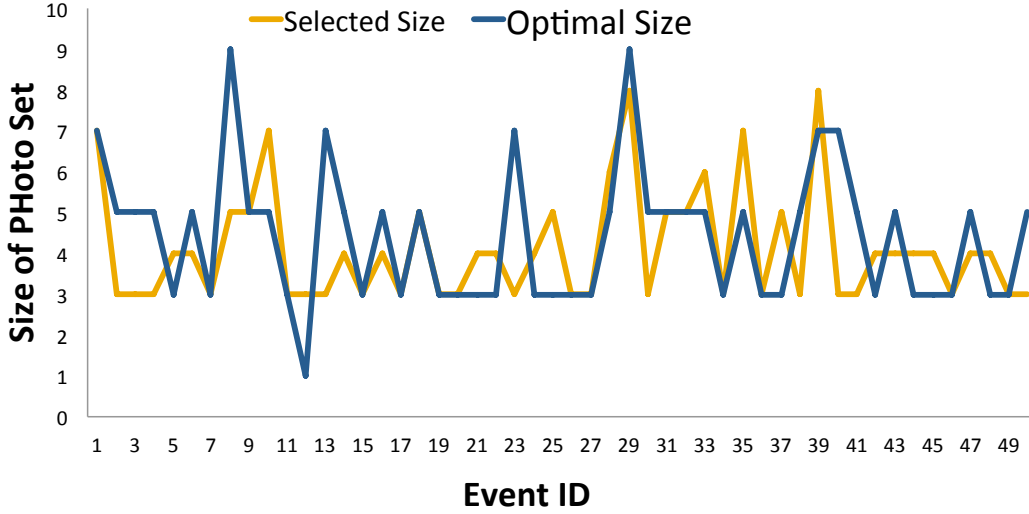


Figure 3.10. Evaluation of image set size selection: sizes generated in automatically (yellow dots) and sizes selected manually by considering each event’s perceptual quality and smallness of size(blue dots).

iteration, the following condition should be satisfied:

$$\frac{f(S_k \cup \{p\}, e) - f(S_k, e)}{f(S_k, e) - f(S_{k-1}, e)} \geq \sigma \quad (3.18)$$

Intuitively, we will stop the iteration of the greedy algorithm if the relative gain in perceptual quality by adding one more image is small.

With the 50 events, we conducted an experiment with our proposed method and generated the best size for each event. Here, we set the threshold $\sigma = 0.5$. The size automatically computed by our method is called the selected size. We also select a size from five sizes (1, 3, 5, 7, 9) in the image set generation experiment that has a relatively high perceptual quality for a small size and call it the optimal size. A comparison of these two sizes is shown in Figure 3.10.

From this result, we can see that the general trend between selected size and optimal size is very similar. It demonstrates that our method can give an approximate size by balancing both perceptual quality and size smallness.

3.7 Conclusion and Future Work

In this research, we first proposed a method to achieve cognition-aware summarization of images presenting events from the perspective of viewers rather than image takers. To improve the perceptual quality of an image set, which is defined as the accuracy and quickness of a users recognition of an event from the image set, we focused on preventing misrecognition of neighbor events, which is different from the work in traditional research on image sets. The reasons for highly possible misrecognitions were analyzed, and three criteria were raised to measure the degree of preventing them. A greedy algorithm was then applied to generate image sets by maximizing the objective function that combines the three criteria. We compared the performance of Visual-Rank and our approach. The results showed that our proposed approach improved the perceptual quality in different sizes of image sets.

Moreover, we analyzed the relationship between the perceptual quality of image sets and event types considering time and location. The results indicated that time is easier to express with images than location. Additionally, events that are not time- or location-specified can be better perceived in images by users. We also conducted an experiment to find the optimal sizes for image sets of events using our proposed method, and the resulting image set sizes were compared with manually selected optimal sizes. We found that our method was able to give an approximate optimal size under the premise of high perceptual quality and small image set size.

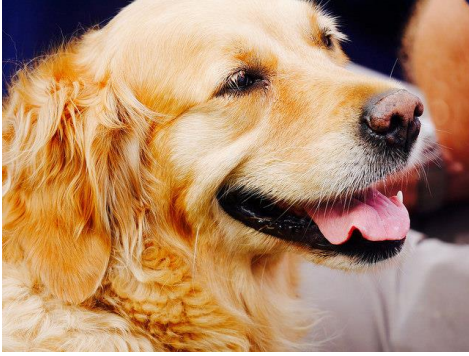
Additional work is needed to address this problem more comprehensively. Although we have considered different factors of an event in the analysis, for example, time and location, we did not make them distinct when generating neighbor events, which will be part of our future work. Moreover, these factors should be taken into consideration when analyzing the relationships between events.

LEARNING SUBJECTIVE ADJECTIVES FROM IMAGES

4.1 Introduction

Understanding *subjective adjectives* (e.g. attributes and sentiment) from visual contents has attracted considerable attentions recently [63, 64, 65, 66, 67]. This problem is more subjective and holistic compared with other image processing problems, including object detection, scene categorization, and textual analysis, and requires much more knowledge about human perception and affection. In natural language, subjective adjectives are defined as *adjectives that express opinions and evaluations* [12]. Learning subjective adjectives from images would enable machines to better understand the connection between visual contents and human impression, and better simulate human understanding and affection.

Although a lot of efforts have been made to learn subjective adjectives mainly with supervised learning approaches (e.g. [64, 63]), it is difficult to apply the existing approaches to image searching; one of the typical applications of subjective adjective understanding. The primary reason is *the lack of training data for a wide variety of noun-subjective adjective pairs in an image search*. As seen from examples in Figure 4.1, the expression of a subjective adjective in images can be different for objects appearing in them (for example, “happy” is expressed with an opening mouth and a



(a) relevant image of “happy dog”



(b) relevant image of “happy girl”

Figure 4.1. Two relevant images with the same “subjective adjective” but different “nouns”.

hanging tongue for dogs, while it is expressed with the rising radian of the mouth for girls). Thus, typical datasets for learning adjectives (*e.g.* [67, 65]) consist of triplets of a noun, an adjective, and a set of images that (1) include objects indicated by the noun and (2) evoke a feeling of the quality of the adjective. Despite a large number of possible noun-adjective pairs that can be used as search queries, which could be much larger than nouns used in object recognition problems, the size of the existing datasets is limited due to a tremendous cost required for human assessment. Therefore, the existing supervised learning paradigm cannot cover many of the potential noun-adjective pairs, and cannot be used for simple image search applications.

In this work, we propose a method of learning subjective adjectives that can be used without images labeled by human annotators, by exploiting results from existing image search engines as *weakly-labeled* data. Given an “subjective adjective noun pair” (ANP) query, we retrieve images from an image search engine by inputting the ANP query and use them as *pseudo-relevant* images that can include both *truly-relevant* and *truly-irrelevant* images. We also obtain *pseudo-irrelevant* images by inputting only the noun in the ANP query. Since the pseudo-relevant and pseudo-irrelevant image sets can contain irrelevant and relevant images, respectively, the key challenge in this work is how to effectively learn subjective adjectives from such weakly-labeled data containing labels with a lot of errors.

To address the challenge concerning weakly-labeled data, we propose a *pairwise stacked convolutional auto-encoder* that can learn discriminative features from pseudo-

relevant and pseudo-irrelevant image sets, and can effectively distinguish truly-relevant and truly-irrelevant images in weakly-labeled data. Unlike conventional stacked convolutional auto-encoder [18], the pairwise stacked convolutional auto-encoder learns a dominant difference shared by a majority of the pairs of a pseudo-relevant and a pseudo-irrelevant image, and represents each image in a feature space where most of the pairs exhibit similar differences. We can then effectively distinguish truly-relevant images from truly-irrelevant ones in this feature space by assuming that a pair of images that exhibit the dominant difference is a pair of a truly-relevant and a truly-irrelevant image.

We conducted experiments with images from Flickr to evaluate the effectiveness of our approach.

Three contributions of this work are summarized as follows:

1. We addressed the problem of the lack of training data for an image search with ANP queries by utilizing existing image search engines as resources for weakly-labeled data.
2. We proposed the pairwise stacked convolutional auto-encoder that identifies a dominant difference between pseudo-relevant and pseudo-irrelevant image sets, and can learn discriminative features for truly-relevant and truly-irrelevant images.
3. We conducted experiments with images from Flickr and demonstrated the effectiveness of our approach with and without labeled data.

The rest of the work is organized as follows. We introduce studies that are related to subjective adjective (sentiment, attribution, emotion, *etc.*) analysis from images and feature representation in Section 4.2. In Section 4.3, we define some of the terminologies used as well as the problem of this research. In Section 4.4, we introduce our framework of pairwise stacked convolutional auto-encoder for learning discriminative features for subjective adjectives and how we use the learned feature to rank images. In Section 4.5, we describe the experiments we conducted with images from Flickr to evaluate the effectiveness of our approach. Finally, we present our conclusion in Section 4.6.

4.2 Related Work

The research on adjective analysis from images can be divided into two main parts in case of visual computing technologies: handcrafted feature based and deep learning feature-based approach.

In tradition, researchers in visual computing try to design handcrafted features to represent images for different purposes. Related works to adjective analysis are sentiment analysis [67], aesthetic evaluation [68], interestingness prediction [69], *etc.* . Borth et al. has made a significant progress in adjective analysis by proposing a mid-level representation framework upon psychology and folksonomies. They also provide a concept detector library based on their ontology.

Recently, deep neural network is widely used in learning robust features from a large number of images [70, 71]. Although handcrafted features can better convey the actual features in a meaningful and intuitive way, deep neural network outperforms with powerful effectiveness. For example, Narihira et al. [64] succeeded in building a visual sentiment ontology from visual data and respects visual correlations along adjective and noun semantics with a factorized CNN model. Jingwen et al. [63] proposed a deep coupled adjective and noun neural network for visual sentiment analysis. However, all those methods that try to learn good features about adjective require a large number of labeled training images and the adjectives (either sentiment or attribute) are limited to the labels in the training dataset. In the context of image search, it is unrealistic to include all possible “adjective” and “noun” combinations, not to mention many labeled images for all the queries.

4.3 Preliminaries

We will define some terminologies used in this research and introduce the problem definition in way of a formula in this section.

4.3.1 Terminology Definitions

To better clarify our research, we will use two different terminologies to indicated the images’ relevance to ANP query: “truly relevant” and “pseudo-relevant”. Truly relevant images are defined as:

Definition 10 (Truly relevant images of an ANP query). *Images that specifically contain exactly object(s) of “noun” and are able to impress viewers with the quality of “adjective”.*

Search results of ANP queries are defined as pseudo-relevant images since they are not all truly relevant images for the three reasons we explain in Section 4.1.

Definition 11 (Pseudo-relevant images of ANP query). *Search result images of ANP query from existing image search engines (e.g. Flickr image search).*

Since our focus in this research is features that are discriminative to “adjectives”, we define truly irrelevant images to an ANP query as:

Definition 12 (Trully-irrelevant images of ANP query). *Images that include the “noun” but cannot impress users of the “adjective”.*

Similarly, pseudo-irrelevant images are defined as:

Definition 13 (Pseudo-irrelevant images of ANP query). *Search results of only “noun” query from existing image search engines (e.g. Flickr image search).*

4.3.2 Problem Definition

In this research, our target is to learn discriminative features for adjective from images of ANP query and then use the learnt features to improve image ranking. An “adjective noun pair” (ANP) query is denoted as $q = \langle a, n \rangle$, where a is an adjective and n is a noun. We use $P = S(\langle a, n \rangle) = \{p_1, p_2, p_3 \dots\}, |P| = m$ to denote top m search result images of “adjective noun” query in image search engines (e.g. Flickr image search) and $Q = S(\langle n \rangle) = \{q_1, q_2, q_3 \dots\}, |Q| = m$ to denote top m search result images of “noun” query. Here P is defined as pseudo-relevant image set and Q is pseudo-irrelevant image set. Have these above as input and $k, k \in \mathbb{N}$, we aim to get output as $O = \{(o_1, 1), (o_2, 2), \dots\}, o_i \in P, |O| = k$.

With an ANP as the input, we can find pseudo-relevant image set P and pseudo-irrelevant image set Q (suppose they are also regarded as inputs). The number of top rankings is also given, denoted as k . Our target is to rank images in P to get the top k images in a sequence based on these images’ relevance to the input ANP query.

4.4 Approach

In this section, we will explain how we learn discriminative features of adjective from images and how these features are used to measure relevance of images to ANP queries.

As we have defined in Section 4.1, an image is relevant to an ANP query when the content of the image includes exactly the “noun” (e.g. sky, cat, people) in the query and we can feel the quality of the “adjective”(e.g. blue, cute, happy) from the image as well. Intuitively, the problem of measuring the relevance of an image to an object (“noun”) is similar to object recognition problem. However, when object and adjective are combined to be measured, the problem becomes much more complicated, because features that make an image relevant to an ANP query depend on both the “adjective” keyword and the “noun” keyword. Traditional training ways that try to learn useful features with supervised methods require a large number of images with ground truth labels. However this is unrealistic for image search problems. Thus, we propose to apply unsupervised approach to learn discriminative features of adjectives for a certain object, and then use these learnt features to estimate relevance of an image to the ANP query.

By applying unsupervised learning method to pseudo-relevant images, such as s-tacked convolutional auto-encoder, it is able to learn representative feature for both “adjective” and “noun”. However, features of “noun” are usually more significant than features of “adjective”. The reason is that discriminative features of “adjective” are usually very subtle [72]. Among all the representative features of ANP query, it is difficult for us to distinguish the discriminative ones for “adjective”. For example, hanging tongue is typical feature for “happy dog” while it will be easily to be dismissed since the shape of mouth is more significant for most images of “dog”. For this reason, we propose to compare pseudo-relevant images with pseudo-irrelevant images to better learn discriminative features for “adjective” of a certain “noun”. Our assumption is that:

Assumption 2. *Discriminative features that help add the quality of “adjective” are similar for one object (“noun”) in certain dimensions.*

With this assumption, we can know that differences that represent discriminative features are similar while differences of other features are not similar. As a result, we can learn the discriminative features by comparing pseudo-relevant images and

pseudo-irrelevant images.

The main approach consists of three parts:

1. Make image pairs that consist of one image from pseudo-relevant images and another from pseudo-irrelevant images,
2. Learn discriminative feature to represent differences between truly relevant images and truly irrelevant images from image pairs with our proposed pairwise stacked convolutional auto-encoder,
3. Use the learnt discriminative features to rank images.

4.4.1 Image Pair Construction

In order to decrease the side effects of many noisy differences between two images, we first conduct a image pair selection from the pseudo-relevant images and the pseudo-irrelevant images. Since the discriminative features we aim to find are more about “adjectives”, objects play a less important role in the difference between a truly relevant image and a truly irrelevant image.

We utilize 16-layer deep neural network [71] to detect objects for all the images in the pseudo-relevant and pseudo-irrelevant images. The algorithm to construct our image pairs from two sets of images is shown in Algorithm 2.

Algorithm 2 Image Pair Construction Algorithm

Require: Pseudo-relevant images: $P = \{p_1, p_2, p_3 \dots\}, |P| = m$;

Pseudo-irrelevant images: $Q = \{q_1, q_2, q_3 \dots\}, |Q| = m$;

Top ten detected objects in each image: $R(p) = \{r_1, r_2, r_3 \dots\}, |R(p)| = 10$ for $p \in P \cup Q$.

Ensure: A set of image pairs: $\{(p_i, q_j)\}, p_i \in P, q_j \in Q$.

- 1: $T = \{\}$
 - 2: **for** each $p_i \in P$ **do**
 - 3: $q^* = \underset{q \in Q}{\operatorname{argmax}} \operatorname{Sim}(R(p_i), R(q))$;
 - 4: $T = T \cup \{(p_i, q^*)\}$;
 - 5: $Q = Q - \{q^*\}$;
 - 6: **end for**
 - 7: **return** T
-

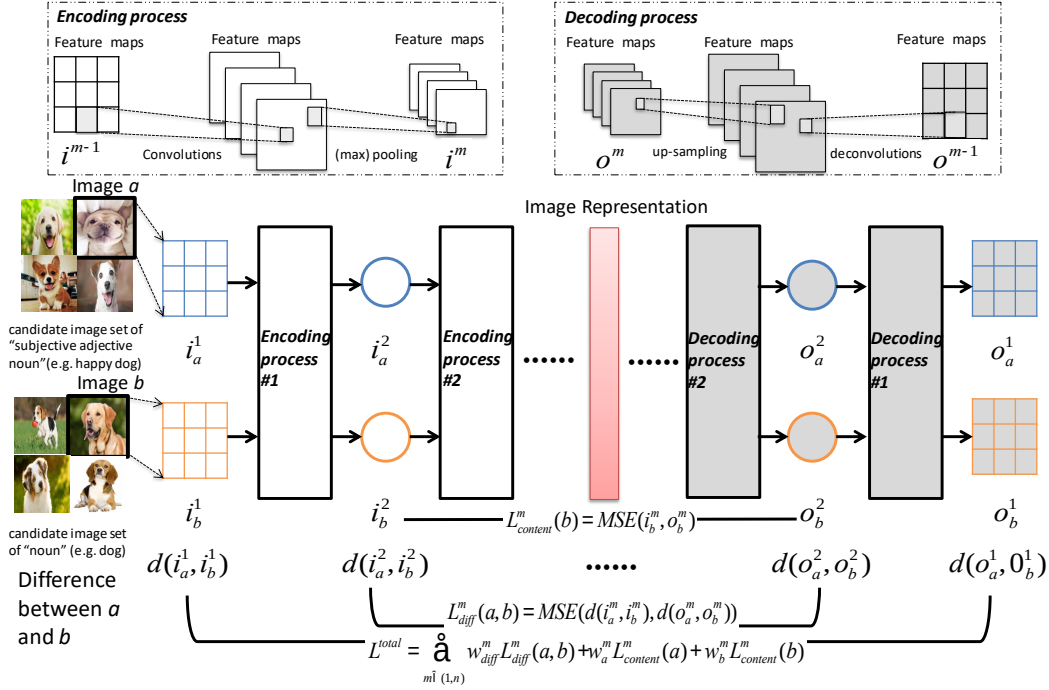


Figure 4.2. The framework of pair-wise stacked convolutional auto-encoder architecture.

Similarity of two sets of object is simply defined as the cosine similarity since the detected objects are among a certain range of classes (1000 classes in ImageNet [73]).

4.4.2 Pair-Wise Stacked Convolutional Auto-Encoder

As we have explained, without a large number of labeled images for image search problems, we apply unsupervised learning method to learn representative features for ANP. The overall architecture we use is stacked convolutional auto-encoder that is used to learn representative features of images. The encoder-decoder paradigm is used in many unsupervised feature learning methods, such as Predictability Minimization Layers [14], Restricted Boltzmann Machines (RBMs) [15] and auto-encoder [16]. The basic idea of auto-encoder is to learn a representation model by making the reconstructed output as similar as the input. The input is first fed to the encoder which produces a lower dimensional space (e.g. feature vector). Then the decoder module reproduces the input from the dimensional space. The encoder and decoder are trained to minimize the average reconstruction error. To deal with 2D image structure with auto-encoder and

reduce redundancy in the parameters brought by global features, convolutional auto-encoder (CAE) is proposed [18]. The weights are shared among all locations in one feature map of a channel and the reconstruction is a linear combination of basic image patches based on the latent code. Deep networks can be trained by building several auto-encoder in a layer-wise way [19] and it is applied in many recent researches [74] [75].

Since our target is more about the discriminative features for the “adjective” of a certain “noun” and they are indicated by differences between image pairs, we propose to modify the auto-encoder into pairwise way. Figure 4.2 shows the architecture of our proposed pairwise stacked convolutional auto-encoder.

Suppose we have one image pair (a, b) , $a \in P, b \in Q$. Both images a and b are passed through the network. The whole network consists of two main parts: encoding part in the first half and decoding part in the second half. The encoding part and the decoding part have the same number of processes in a stacked way. In other words, the first encoding process is corresponding to the last decoding process. The output of lower process serves as the input of next process. As we can see from Figure 4.2, suppose we have three processes in the encoding part: *Encoding process #1*, *Encoding process #2*, and *Encoding process #3*. The three corresponding decoding processes are: *Decoding process #3*, *Decoding process #2*, and *Decoding process #1* in sequence. We use the same number to indicate corresponding encoder and decoder.

The upper part of Figure 4.2 explains the detailed workflow of each encoding process and decoding process. Each encoder consist of a convolutional layer to map the input to several feature maps with different kernels (convolutional matrix) and a max-pooling layer for spatial down-sampling. The decoder includes an up-sampling layer and a deconvolutional layer. Suppose in the m -th encoding process, the input of image a is denoted as i_a^m and we have latent feature maps of number H . For the input i_a^m of k -th feature map ($0 < k \leq H$), the representation is computed as:

$$y_a^{m(k)} = \sigma(i_a^m * W^{m(k)} + b^{m(k)})$$

. Here σ is an activation function and $*$ denotes the 2D convolution. The bias $b^{m(k)}$ is broadcasted to the whole map. The output of the 2D convolution is then applied with max-pooling to down-sampling the latent representation to improve filter selection and avoid overfitting by taking the maximum activity within input feature maps. The output of m -th encoding process is then used as input of $m + 1$ -th encoding process. In its corresponding m -th decoding process, up-sampling is first done to restore the max-

pooled features. Suppose after unpooling process, for k -th feature map ($0 < k \leq H$), we have up-sampled output of last decoding process denoted as $z_a^{m(k)}$. The reconstruction is obtained with:

$$o_a^m = \sigma\left(\sum_{k \in H} z_a^{m(k)} * \tilde{W}^{m(k)} + c\right).$$

$\tilde{W}^{m(k)}$ denotes the flip operation over both dimensions of the weights in the k -th feature map of m -th encoding process.

In usual convolutional auto-encoder, mean squared error (MSE) between the input and reconstructed output is used to measure the cost function that is to be minimized. As in the standard neural networks, the backpropagation algorithm is applied to compute the gradient of the cost function with respect to the parameters. As a result, the representative features are learnt to reconstruct the input image as well as possible.

In our pair-wise auto-encoder, we are supposed to find the representative differences between sets of image pairs and we hope our network can well reconstructed these differences. Thus, in addition to computing cost function with reconstructed output and original input, we also compare differences of the reconstructed output with differences of original input. As we can see from Figure 4.2, for two images a and b in the image pair, suppose the input of the m -th encoding process are denoted as i_a^m and i_b^m respectively. We use $d(i_a^m, i_b^m)$ to define the difference of these two images' input before passing them into the m -th encoding process through our network:

$$d(i_a^m, i_b^m) = i_a^m - i_b^m.$$

Similarly, we use o_a^m and o_b^m to represent the feature representation (reconstructed output) after m -th decoding process for image a and image b through the network, and their difference are denoted as $d(o_a^m, o_b^m)$. We then define the squared-error loss between the two differences in k -th encoding process and k -th decoding process:

$$L_{\text{diff}}^m(a, b) = \text{MSE}(d(i_a^m, i_b^m), d(o_a^m, o_b^m)).$$

For image a and image b , Mean Squared Error is also computed to measure the loss between input and reconstructed output in m -th process:

$$L_{\text{content}}^m(a) = \text{MSE}(i_a^m, o_a^m),$$

$$L_{\text{content}}^m(b) = \text{MSE}(i_b^m, o_b^m).$$

Suppose we have n encoding processes and n decoding processes, the total loss is weighted sum of mean square errors in all corresponding encoding-decoding processes:

$$L^{total} = \sum_{m \in (1, n)} (w_{\text{diff}}^m L_{\text{diff}}^m(a, b) + w_a^m L_{\text{content}}^m(a) + w_b^m L_{\text{content}}^m(b))$$

where w_{diff}^m is weighting factor to indicate the contribution of two image' difference in m -th process to the total loss. w_a^m and w_b^m indicate the importance of reconstruction of image contents in m -th process for image a and b .

The backpropagation algorithm is applied to compute the gradient of the error function with respect to the parameters.

4.4.3 Image Ranking with Learnt Features

After optimization of lost function in our pairwise stacked convolutional neural network, a series of parameters are learnt in the network. With these learnt parameters, we are able to encoder an input image to some feature maps that can represent the input images with most representative features. In the third step, we rank the pseudo-relevant images with VisualRank [76] by defining the similarity between two images using these encoded features maps. VisualRank will rank images that are have most similar feature maps to other images in the tops.

The formula of VisualRank is shown as follows: given n images, VR is recursively defined as

$$VR = dM^* \times VR + (1 - d)p,$$

where $p = [\frac{1}{n}]_{n \times 1}$. M^* is the column normalized adjacency matrix M , where $M_{i,j}$ is the similarity between image p_i and p_j , which is computed by Euclidean distance between visual features of two images. Visual features of each image is the encoded feature maps after applying the trained encoders in our pairwise stacked convolutional auto-encoder.

4.5 Experiment

We evaluate the proposed pairwise stacked convolutional auto-encoder on ten queries with images we crawled from Flickr.

4.5.1 Datasets

Table 4.1. The queries we used in the experiment. (Ratio A: ratio of truly relevant images in pseudo-relevant images, Ratio B: ratio of truly relevant images in pseudo-irrelevant images)

Query	Ratio A	Ratio B
happy dog	0.785	0.18
tiny flower	0.688	0.3
clear sky	0.565	0.2
ancient city	0.865	0.2
falling snow	0.735	0.25
warm water	0.425	0.075
happy kids	0.83	0.3
dry flower	0.81	0.055
fluffy clouds	0.899	0.675
fresh flowers	0.78	0.6

In the experiment, because of the restricted images crawling from the search engines (not allowed to crawl or a very limited number of permission), we decided to use existing dataset that used in [67]. One advantage of using this dataset is that with the labels for each images, we do not need to spend extra cost to evaluate whether an image is relevant to a query or not in the evaluation phase. The images in the dataset are from Flickr and the dataset include 1553 ANPs (Adjective Noun Pairs) with their images. In order to make our dataset, we clustered all the ANPs based on nouns. The we selected ten queries (ANPs as called in their research) with nouns that have many adjectives in the cluster. We also considered the number of images for the queries to make sure that each query have more than 1000 images. Table 4.1 lists all the queries we used in the experiment.

To better simulate the ratio of truly relevant images in the pseudo-relevant image set and pseudo-irrelevant image set as in the real search engines, we conducted a survey of these ten queries in web image search engines (Google and Flickr). For each query, we surveyed the ratio of truly relevant images to the query in the top 200 result images with both pseudo-relevant images (results of the subjective adjective noun query) and pseudo-irrelevant images (results of the noun query). We took the average number as the simulation ratio as showed in Table 4.1. For each query, images of the query are

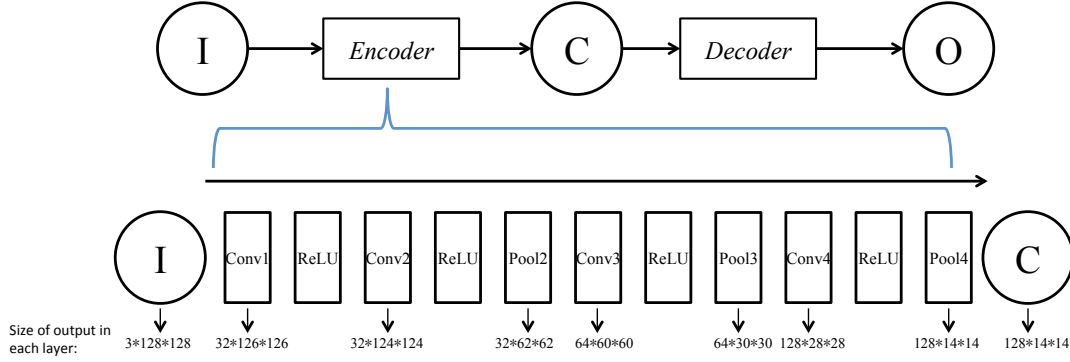


Figure 4.3. Structure of neural network in experiment.

treated as truly relevant images and images of other ANPs with the same noun are treated as truly irrelevant images (we also filtered some ANPs that have very similar subjective adjectives, such as “excited kids” for “happy kids”). The pseudo-relevant image dataset and pseudo-irrelevant image dataset were constructed by adding images from the truly relevant images and truly irrelevant images with their numbers fit the ratio we surveyed in real image search engines.

4.5.2 Network Structure

In the experiments, after experimental trial, we set four encoders and four decoders in our pairwise stacked convolutional auto-encoder architecture. Figure 4.3 shows the of each layer during encoding processes.

All the input images are first resized to 128×128 in three channels (RGB) before passing to the network. For the size of filters, we follow the idea of VGG net proposed in [71], We use two 3×3 filters for the first two convolutional layers. Two ReLU layers are followed after each convolutional layer and a 2×2 max-pooling is done after these two convolutions. Smaller filters help retain a lot of original pixel information in the input since some differences between truly relevant images and truly irrelevant images rely on those small details, and two continual convolutional layers can simulate a larger filter that are used in many other networks. Max-pooling is also applied in the third and fourth encoding processes. Size of input and output in each layer is also shown in Figure 4.3. After several empirical tests, we finally set 32 filters for the first two convolutional layers and double the number in the third and fourth convolutional layers. As a result, the output of all the encoding processes as well as the input of

decoders is sized to $128 \times 14 \times 14$.

Each time for training, we have a batch of image pairs passing to the network together to get the optimized parameter and we set the batch number to 64. For all the training image pairs, we have 20 times forward and backward processes (number of epoch is set to 20).

To determine the layer in which input should be well reconstructed, we do experiments on each layer by setting weights on layer to 1 and weights on layers to 0. The most above layer turns out to be most proper with quickest converging and better reconstruction. This can be also well explained in a intuitive way. The higher layers are more about abstract content while the lower layers are responsible for pixel-level reconstruction. In our research, since we aim to find common features in most images that are representative for ANP query and we do not care about pixel-level reconstruction, higher level features with more abstract contents should be reconstructed well. Thus, we set $w_{\text{diff}}^4 = 1$ and $w_a^4 = 1$. Weights in all other layers are set to 0. In addition, we find that reconstruction of pseudo-relevant images among the image pairs is enough to address the content reconstruction, and for better comparison with single s-tacked convolutional auto-encoder, we set weights of reconstructing pseudo-irrelevant images' content to 0. Single stacked convolutional auto-encoder is trained by setting $w_a^4 = 1$ and all other weights to 0.

4.5.3 Result

Table 4.2. Result of our approach and the precision of top 200 in image search engines for two queries.

Query	Precision@200*	Accuracy of ours
happy dog	0.565	0.617
clear sky	0.785	0.802

* the mean precision of top 200 in image search engines (Google image and Flickr)

Table 4.2 shows comparison of our approach and the mean precision of top 200 in image search engines (Google image and Flickr) for two queries. We can see that our approach could slightly outperform the current image search engine when the query is a “subjective adjective noun” query.

We consider much more space for improvement in this research. In the future, we will take consideration of similar subjective adjectives when getting pseudo-relevant images and pseudo-irrelevant images. Images in the dataset we use are not really truly relevant images and truly irrelevant images. In that case, we consider to make our own dataset that can perfectly match our research goal, such as sequence of images for the same objects. Parameter is a very important factor to influence the performance of deep neural network and we will need more trials to adjust them to make better performance. Moreover, we will try to get visual representation of the learnt features to have a better and intuitive understanding of what we have learnt with the network.

4.6 Conclusion

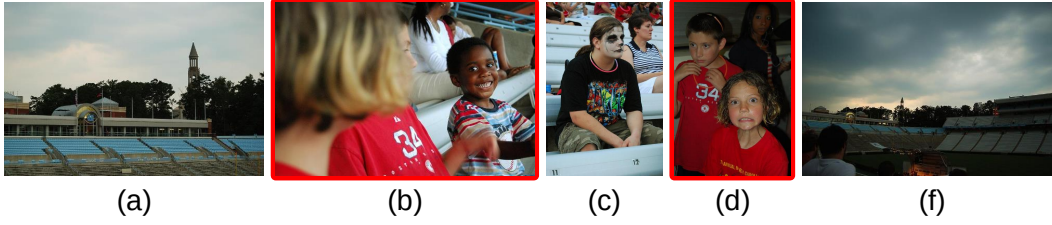
In this work, we propose to solve the problem of estimating relevance of images to “subjective adjective noun” queries by first learning discriminative features of “subjective adjective” from images with unsupervised deep convolutional auto-encoders and then learn to measure the relevance. We propose pair-wise stacked convolutional auto-encoders to find discriminative features that can represent differences between relevant images and irrelevant images. We show our conducted experiment and the result is compared with precision of some image search engines. Finally we make a discussion according to the result and we list some future plans.

VISUAL STORYTELLING

5.1 Introduction

Recent years, we have seen a bursting number of researches in bridging the gap between vision and language. Driven by the availability of large scale of pairing image and natural language descriptions and successful use of recurrent neural network (RNN), encouraging progress has been made in language generation from images [6, 7, 77]. In this work, we tackle the problem of generating a story that consists of several sentences from a sequence of images, i.e., visual storytelling. Compared with image captioning and paragraphing, visual storytelling is a more subjective task that requires an overall understanding and connection of all images and aims to generate sentences with consistent semantics.

Visual storytelling has been explored by several researches in recent years [78, 8, 4, 1, 79]. Compared with image description generation, visual storytelling is facing with two problems. Firstly, the ordered sequence of images guides the content of the stories, which means the same image will correspond to different contents when it occurs in different sequences of images. Secondly, due to the fact that stories come from the integration of visual contents and human understanding, human's different perception will lead to different stories even for the same sequence of images. Fig. 5.1 shows an example of different stories annotated by different user for the same sequence of images. Most current related works deal with the first problem and they usually



Annotator 1:

- (a) On Saturday we arrived at the stadium. we had waited all week for this adventure.
- (b) [male] was **surprised** to see a school friend of his.
- (c) It 's strange the things one will see in public sometimes. The kids wanted to take this picture.
- (d) The kids were really **excited** . they had talked about it all week.
- (e) We became fearful when the sky began to darken. We had not realized that there were weather warnings.

Annotator 2:

- (a) it was a nice evening to visit the stadium with the family.
- (b) The children were **excited** and ready to make friends with fellow spectators.
- (c) One eager young man painted his face in celebration.
- (d) As the evening progressed, the children grew **giddy** with anticipation.
- (e) A darkened sky and a commotion in the field meant the festivities were about to begin.

Figure 5.1. Example of stories annotated by different users for the same image. For image (b), annotator 1 reads **surprising** from the kid’s face while annotator captures **excitement**. For image (d), the kids are **excited** for annotator 1 while they are **giddy** for annotator 2. Sentences for these two images contain different contents with different emotion interpretation.

ignore the fact that human-level stories are resulted by not only visual contents, but also other factors. Thus they result in stories with general sentences that are suitable for many images. In this work, we mainly focus on the second problem. There are many factors that influence our understanding of an image sequence. From the perspective of human perception and affection, we consider emotion as one of the key factors. Given the example of Fig. 5.1, different emotions for the image (b) and (d) cause different sentences for these images and also affect the trend of whole story.

To generate a human-like story from an image sequence, we are facing with the following three challenges. First of all, compared with existing works of generating stories from image sequences, we have emotion as an additional input to the story generation. Secondly, to simulate human process of telling a story of an image sequence, we require an image sequence corresponding to several different emotion sequences. In addition, emotion of one image depends on not only visual content, but also its contextual images in a sequence. Thirdly, as a story generated from an image sequence

conditioned by emotion, we require the generated story to satisfy image-relevance, emotion-consistency and story-likeness.

To address the above challenges, we propose a coupled-RNN visual story generator with image stream and emotion as input and the generated stories are further optimized by policy gradient. Two discriminator networks aims to guarantee the generated sentences are relevant to each corresponding image and in accordance with story language style, and one similarity function is computed to measure consistency of generated story and input emotion. The two discriminators and emotion similarity function are jointly used to provide rewards for story generation approximation. To predict several emotions from one image, we propose to utilize conditioned variational auto-encoder (CVAE) [80] to generate diverse but realistic emotions. In order to make the predicted emotions coherent in an image stream, we recurrently connect CVAEs of each emotion prediction model to sequentially update the predicted emotions. we conduct experiments on visual storytelling (VIST) dataset [8]. The generated stories are evaluated in both objective and subjective ways. We define automatic evaluation metrics in terms of image relevance, emotion relevance and expressiveness. User studies are also conducted concerning these three metrics. Turing test is performed to test the human-likeness of our generated stories. The contribution of this work can be concluded as follows:

- We propose to introduce emotion as an important factor to generate story from image stream. To the best of our knowledge, this is the first attempt to put forward emotion for visual story generation, which enables a machine to generate human-level stories.
- We incorporate an emotion prediction model which is able to predict diverse and coherent emotions and a coupled-RNN story generator with image stream and emotion as input, in which two discriminators and a similarity function provide rewards for measuring image relevance, emotion relevance and story style.
- We conduct extensive experiments to demonstrate the effectiveness of our approach compared with several baselines in both objective and subjective evaluation metrics.

5.2 Related Work

There are many studies conducted on generating sentence(s) from images. We will review them based on two categories: visual description generation and visual storytelling.

5.2.1 Visual Description Generation

Visual description generation (image captioning and paragraphing) aims to find or generate sentence(s) to describe one image. It is first researched as a retrieval problem so that sentences with similar semantic meaning will be searched [81, 82]. The problem of search-based generation is that it cannot provide accurate sentences for images. Template filling method is thus proposed to overcome this problem [83]. Recently, with the development of deep neural network, integration of convolutional neural network (CNN) and recurrent neural network (RNN) is boosting the sentence generation research for readable human-level sentences [84, 85, 6, 7, 86]. Later on, generative adversarial network (GAN) is utilized to improve generated sentences for different problem settings [77, 26]. However, as we have addressed, the target of image description generation is to use sentence(s) to describe factual visual content while story is a combination of visual contents and human subjective perception (emotion).

5.2.2 Visual Storytelling

Visual storytelling is a rather new topic but has attracted many attentions. Generating several sentences for the purpose of storytelling is more challengeable than visual description for one image. Relationship between different visual contents need to be considered to form a good story and sentences for a story have to be coherent. Similar to visual description researches, early works mainly focus on search-based method to retrieve the most suitable sentence combination for an image sequence [78]. [4] proposes a skip Gated Recurrent Unit to deal with semantic relation between image sequence and generated sentences. Then methods leveraged by image captioning, especially CNN-RNN framework is extended for story generation [8]. Recently, we have seen some works that utilize reinforcement learning and generative network for better story generation [1, 79, 87]. Though topic is introduced in [87], existing works still lack of subjective perception of human when making stories, which we first introduce in this paper.

5.3 Approach

Our model can be considered as an encoder-decoder framework, implemented with a hierarchical recurrent neural network structure. The encoders can be considered in two main parts: one being a simple story generator, the other being our novel sequential emotion generator.

In addition, we apply reinforcement learning with emotion reward to our model.

5.3.1 Overview

Given a photo sequence, we first extract the global features of each image with a VGG-16 model. The outputs from the last fully connected layer (fc7) is utilized. We then feed the image features to both the story generator and the emotion generator.

The story generator is an RNN with gated recurrent unit (GRU) as its cell. It is used to generate the story feature of each image. At the i -th time step, we feed the GRU the feature of the i -th image as its input, and takes the hidden state of the GRU cell as its output. With the RNN, the coherence among photos is enhanced, which is crucial to story generation. Therefore we consider the output the story feature of the current image.

The emotion generator is a sequentialized conditional generative adversarial network. It generates a creative yet plausible emotion for each image as it not only takes the image as input, but also takes a random noise vector as input, which gives the model creativeness. It is based on the original CGAN. We added a GRU layer in the generator. By adding the additional GRU layer, when generating each emotion, the generator is aware of the previous emotions predicted. Therefore the output contains contextual information as well. The generator is supervised by two discriminators, a relevance discriminator and a consistency discriminator.

The outputs of both generator are concatenated and fed into a decoder RNN. It is a language model that predicts the best possible sentence based on its input, in this case, the story feature and the emotion feature of the current image.

We implemented reinforcement learning. The reward consists of three parts, image-relevance, story-likeness and emotion-consistency. The first two parts, image-relevance and story-likeness are scored by two separate discriminators. The emotion-consistency is measured by comparing the emotion of the story that our model generates with the emotion our model is given.

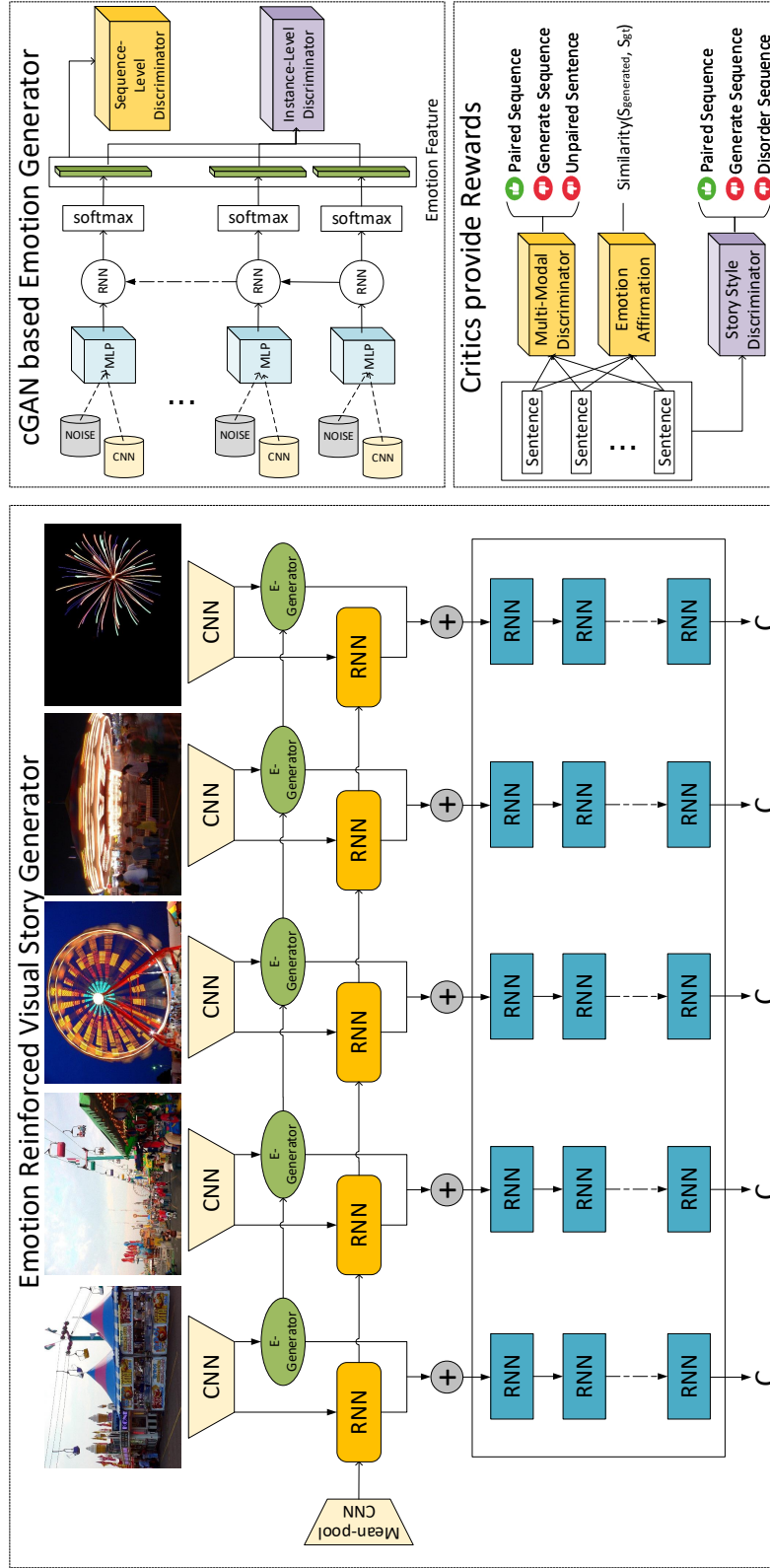


Figure 5.2. Framework of our approach.

5.3.2 Emotion Generator

Our emotion generator is based on CGAN and follows the generator-discriminator structure. The generator is a multi-layer perceptron, followed by a GRU:

It takes two inputs, the image features of the photo sequence and a sequence random noise vector and learns to generate a creative yet plausible emotion vector. The GRU in our generator enables it to generate context-aware emotions.

We implemented two levels of discriminators in our work. One being a instance-level discriminator that measures the image-relevance of the generated emotions, while the other being a sequence-level discriminator that measures the consistency of the generated story sequence.

The instance-level discriminator is composed of a multi-layer perceptron that takes a emotion and a image feature as input, and learns to predict whether the input emotion is the groundtruth emotion paired with the image, a random emotion picked from the dataset or the generated emotion.

The story-level discriminator, on the other hand, is composed of a GRU layer, followed by a multi-layer perceptron. It takes the whole emotion sequence as input, and learns to predict whether the input emotions are a sequence from the dataset, a sequence of randomly picked emotions or a generated sequence.

The generator and the discriminators are jointly trained. Just like the original CGAN, when training the generator, we connect the generator and the discriminators together and fix the discriminator.

5.3.3 Story Generator

We use a GRU as the story generator. The GRU is a type of RNN that defined as:

It contains a hidden state that serves as its memory and updates the hidden state based on the current input, its reset gate and update gate, as well as the last hidden state.

Since the RNN generates the output based on the current input and the previous inputs, the output contains the contextual information of the current image. Therefore, the coherence among photos is enhanced. Because coherence is a crucial part of storytelling, we consider the output of the RNN the story feature of the input image.

5.3.4 Emotion Feature

We incorporate an emotion detection model for emotion extraction. Given a input sentences, it predicts the most possible emoji to express the sentence. The probability distribution on all possible emojis are utilized as the emotion feature in our experiments.

5.3.5 Decoder

Given the predicted story feature and generated emotion feature, the decoder predicts the best possible sentence. We use a RNN language model as decoder, which predicts sentences by predicting each word according to the story and emotion feature, as well as all the previously predicted words.

5.3.6 Reinforcement Learning

We incorporate reinforcement learning in our approach by considering our generator as the agent, and each word picked as an action given the situation. The generated is guided with a reward that consists of 3 parts, image-relevance, story-likeness and emotion-consistency. The first two measurements are judged by two discriminators, an instance-level discriminator that measures the image-relevance and a sequence-level discriminator that measures the story-likeness, as discribed by [1]. The instance-level discriminator is trained to discriminate paired sentences and images from randomly selected sentences and generated sentences, while the story-level discriminator is trained to discriminate real stories picked from the dataset from stories formed with randomly selected sentences and stories generated by our generator.

5.4 Experiment

5.4.1 Dataset and Analysis

We conduct our experiments on the VIST dataset created by [8]. The VIST dataset is created for the task of visual storytelling. It contains 81,743 photos obtained from Flickr website with 20,211 image sequences arranged. Stories for each image sequence are annotated through AMT (Amazons Mechanical Turk). Each sequence contains 5 images and most sequences has multiple annotations.

For preprocessing, we filtered out sequences with images that are no longer available on Flickr, with 40,143 photos and 80,21 sequences remaining for training set, 5,055 photos and 1,011 sequences for testing set. In addition, we tokenized the sentences and filtered out words with occurrence less than 4, creating a vocabulary with 10,698 words.

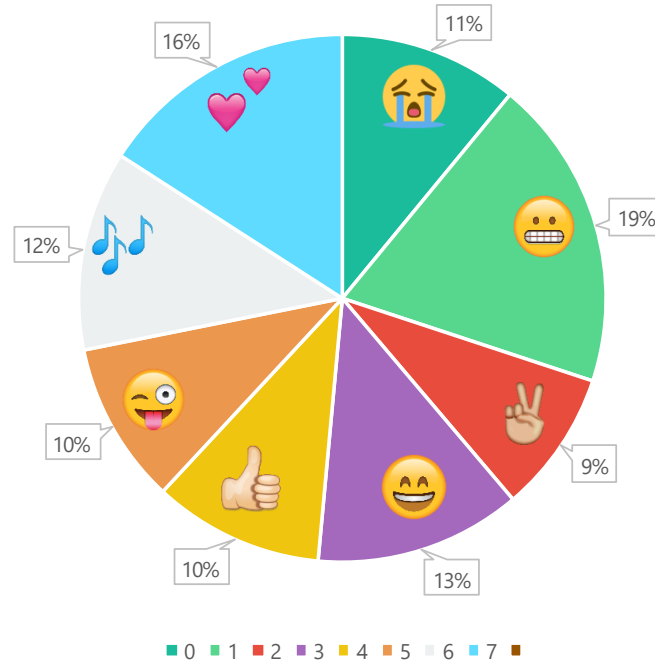


Figure 5.3. Distribution of emotions in the training dataset of VIST. Emotions are extracted by DeepMoji and clustered into 8 types, in which a typical emoji is shown.

To investigate our proposal that emotions play an important role in stories, we make an analysis to sentences in the VIST dataset from two perspectives. Firstly, we check the diversity of emotions among annotated sentences. Figure 5.3 shows the emotion (labeled by emojis) distribution in the training dataset. We can see that different emotions are equally distributed among all the sentences. Secondly, we investigate the emotion difference of sentences for the same image. Suppose for each image, we have n sentences annotated by n users for image i and m different emotions predicted from

these sentences. We compute diversity of emotions for the same image as:

$$D(i) = \log_n \frac{m}{n} + 1, 1 \leq m \leq n, n > 1. \quad (5.1)$$

$D(i)$ is cast to range $[0,1]$. The average diversity of all sentences is 0.72 and this results verifies our assumption that stories annotated by different users will include different emotions.

5.4.2 Experiment Settings

In our experiments, we use the outputs from the fc7 layer of a pre-trained VGG16 model, which has 4096 dimensions, as our image features. The sizes of the hidden states of the story encoder RNN and the language decoder RNN are 1,000 and 1,200 respectively. We utilized DeepMoji [?] to extract emotion features for sentences. The emotion features are of 64 dimensions and are embedded into a space with 200 dimensions.

5.4.3 Compared Methods

To investigate the effectiveness of the proposed methods, we compare the results of our models with four baseline methods. We include several image/video captioning models and the previous state-of-the-art model on story generation. The models are:

Sentence-Concat [6]: a classic method to incorporate the basic CNN encoder - RNN decoder framework on the problem of image captioning. For story generation, we simply concatenate individual outputs for each photo together for the complete story.

Regions-Hierarchical [7]: a hierarchical recurrent neural network that generates several sentences for single images.

SRT [1]: the current state-of-the-art which is the first to incorporate reinforcement learning in the task of storytelling. We test their methods with two settings: **SRT w/o D** and **SRT w/ D** for fair comparison with our model with and without discriminators.

Our Model: to examine the effectiveness of two discriminator and emotion affirmation as rewards, we train our model with two settings. Pretrained model without critics (**Ours w/o critics**) and with all critics (**Ours**).

5.4.4 Objective Evaluation Metrics

For objective evaluation, similar to other visual storytelling researches, we compare the generated stories with reference stories and compute the language similarity with

Method	BLEU-1	BLEU-4	METEOR	CIDEr
Sentence-Concat[6]	38.28	4.17	10.52	8.45
Regions-Hierarchical[7]	34.92	3.70	9.97	6.51
SRT w/o D[1]	40.88	4.49	11.03	9.79
SRT w/ D[1]	43.40	5.16	12.32	11.35
Ours w/o critics	47.0	5.4	11.5	7.0
Ours	47.1	5.5	11.6	7.4

Table 5.1. Automatic evaluation. All scores are reported as percentage (%).

some NLP metrics (e.g., METEOR, BLEU, CIDEr). METEOR and BLEU are evaluation metrics for machine translation whereas CIDEr is designed for image based captioning tasks. The three metrics calculate scores based on the correlation between the generated stories and the ground-truth stories. Note that the relevance here only indicates the relevance between generated stories and reference stories and cannot reflect whether the generated stories are really relevant to image sequence.

5.4.5 Results and Analysis

Table 5.1 shows the evaluation result of our method compared with baseline methods. We can see that our approach achieves the highest score in BLEU-1 and BLEU-4. Figure 5.4 shows some examples of our generated method compared with state-of-art method and ground-truth stories annotated by users. We can see the comparison from words in red and green that our approach is able to generate more expressive and more concrete contents.

5.5 Conclusion

We introduce emotion as an important factor for the purpose of visual storytelling. Our approach incorporate a cGAN based emotion generator and a CNN-RNN based story generator. Image content and emotion are encoded and combined as input for story decoder. Emotion affirmation and two discriminators provide rewards to optimize the generated stories.



State-of-art: The family went to the market for some food. The food was delicious. They food was **great** and everyone was having a **great** time. The beer was **great** and the food was **great**. The man is happy to see that he is going to eat.

Ours: With emotion: Everyone was waiting for the food to begin. We had a lot of food. The man and woman are seen **laughing and conversing** after the meal. The wine was **very good**. The man was seen **doing a great job** on the great day.



State-of-art: The sky was a blue sky and it was clear to be in the sky. The mountain was **beautiful**. The mountain view was **beautiful**. The mountain was **beautiful**. The valleys were where all the people lived.

Ours: The man went on a vacation to the mountains. The mountain was **magnificent**. The mountains were **beautiful**. I took a picture of a small village that was built in location. The mountains were **gorgeous**.



State-of-art: The speaker was presenting with his speech. The speaker is very **motivating**. The speaker is very **motivating**. He was very **happy** to have done a well earned. He was very **happy**.

Ours: The presentation was ready for the conference. The first speaker **was the first to be there** and he **opened the business**. He also **gets a chance to meet the boss**. He was very happy. The speaker is finished and the **award is placed on the table**.

Figure 5.4. Example of stories generated by state-of-art method [1] and our approach.

CHAPTER 6

POETRY GENERATION FROM IMAGE

6.1 Introduction

Researches that involve both vision and languages have attracted great attentions recently as we can witness from the bursting works on image descriptions like image caption and paragraph [26, 5, 7, 6]. Image descriptions aim to generate sentence(s) to describe facts from images in human-level languages. In this work, we take one step further to tackle a more cognitive task: generation of poetic language to an image for the purpose of poetry creation, which has attracted tremendous interest in both research and industry fields.

In natural language processing field, poem generation related problems have been studied. In [88, 89], the authors mainly focused on the quality of style and rhythm. In [90, 89, 91], these works have taken one more step to generate poems from topics. Image inspired Chinese quatrain generation is proposed in [92]. In the industrial field, Facebook has proposed to generate English rhythmic poetry with neural networks [88], and Microsoft has developed a system called XiaoIce, in which poem generation is one of the most important features. Nevertheless, generating poems from images in an end-to-end fashion remains a new topic with grand challenges.

Compared with image captioning and paragraphing that focus on generating descriptive sentences about an image, generation of poetic language is a more challenging problem. There is a larger gap between visual representations and poetic symbols that

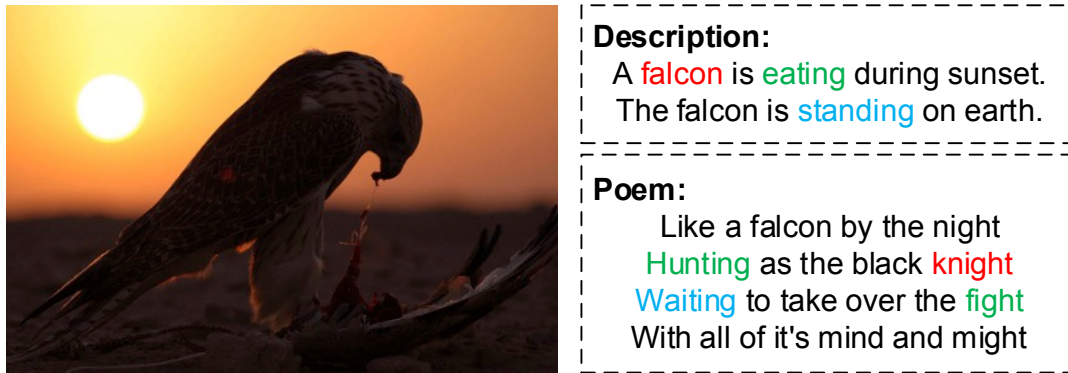


Figure 6.1. Example of human written description and poem of the same image. We can see a significant difference from words of the same color in these two forms. Instead of describing facts in the image, poem tends to capture deeper meaning and poetic symbols from objects, scenes and sentiments from the image (such as **knight** from *falcon*, **hunting** and **fight** from *eating*, and **waiting** from *standing*).

can be inspired from images and facilitate better generation of poems. For example, “man” detected in image captioning can further indicate “hope” with “bright sunshine” and “opening arm”, or “loneliness” with “empty chairs” and “dark” background in poem creation. Fig. (6.1) shows a concrete example of the differences between descriptions and poems for the same image.

In particular, to generate a poem from an image, we are facing with the following three challenges. First of all, it is a cross-modality problem compared with poem generation from topics. An intuitive way for poem generation from images is to first extract keywords or captions from images and then consider them as seeds for poem generation as what poem generation from topics do. However, keywords or captions will miss a lot of information in images, not to mention the poetic clues that are important for poem generation [90, 91]. Secondly, compared with image captioning and image paragraphing, poem generation from images is a more subjective task, which means an image can be relevant to several poems from various aspects while image captioning/paragraphing is more about describing facts in the images and results in similar sentences. Thirdly, the form and style of poem sentences is different from that of narrative sentences. In this research, we mainly focus on *free verse* which is an open form of poetry. Although we do not require meter, rhyme or other traditional poetic techniques, it remains some sense of poetic structures and poetic style language in po-

ems. We define this quality of poem as *poeticness* in this research. For example, length of poems are usually not very long, specific words are preferred in poems compared with image descriptions, and sentences in one poem should be consistent to one topic.

To address the above challenges, we collect two poem datasets by human annotators, and propose poetry creation by integrating retrieval and generation techniques in one system. Specifically, to better learn poetic clues from images for poem generation, we first learn a deep coupled visual-poetic embedding model with CNN features of images, and skip-thought vector features [93] of poems from a multi-modal poem dataset (namely “MultiM-Poem”) that consists of thousands of image-poem pairs. This embedding model is then used to retrieve relevant and diverse poems from a larger uni-modal poem corpus (namely “UniM-Poem”) for images. Images with these retrieved poems and MultiM-Poem together construct an enlarged image-poem pair dataset (namely “MultiM-Poem (Ex)”). We further propose to leverage the state-of-art sequential learning techniques for training an end-to-end image to poem model on the MultiM-Poem (Ex) dataset. Such a framework ensures substantial poetic clues, that are significant for poem generation, could be discovered and modeled from those extended pairs.

To avoid exposure bias problems caused by long length of long sequence (all poem lines together) and the problem that there is no specific loss available to score a generated poem, we propose to use a recurrent neural network (RNN) for poem generation with multi-adversarial training and further optimize it by policy gradient. Two discriminative networks are used to provide rewards in terms of the generated poem’s relevance to the given image and poeticness of the generated poem. We conduct experiments on MultiM-Poem, UniM-Poem and MultiM-Poem (Ex) to generate poems to images. The generated poems are evaluated in both objective and subjective ways. We define automatic evaluation metrics concerning relevance, novelty and translative consistence and conducted user studies about relevance, coherence and imaginativeness of generated poems to compare our model with baseline methods. The contributions in this research are concluded as follows:

- We propose to generate poems (English free verse) from images in an end-to-end fashion. To the best of our knowledge, this is the first attempt to study the image-inspired English poem generation problem in a holistic framework, which enables a machine to approach human capability in cognition tasks.
- We incorporate a deep coupled visual-poetic embedding model and a RNN-based generator for joint learning, in which two discriminators provide reward-

s for measuring cross-modality relevance and poeticness by multi-adversarial training.

- We collect the first paired dataset of image and poem annotated by human annotators, and the largest public poem corpus dataset. Extensive experiments demonstrate the effectiveness of our approach compared with several baselines by using both objective and subjective evaluation metrics, including a Turing test from more than 500 human subjects. To better promote the research in poetry generation from images, we have released these datasets and our code on Github*.

6.2 Related Work

6.2.1 Poetry Generation

Traditional approaches for poetry generation include template and grammar-based method [94, 95, 96], generative summarization under constrained optimization [89] and statistical machine translation model [97, 98]. By applying deep learning approaches recent years, researches about poetry generation has entered a new stage. Recurrent neural network is widely used to generate poems that can even confuse readers from telling them from poems written by human poets [90, 99, 88, 21, 91]. Previous works of poem generation mainly focus on style and rhythmic qualities of poems [88, 89], while recent studies introduce topic as a condition for poem generation [90, 99, 89, 91]. For a poem, topic is still a rather abstract concept without specific scenarios. Inspired by the fact that many poems were created in a conditioned scenario, we take one step further to tackle the problem of generating poems inspired by a visual scenario. Compared with previous researches, our work is facing with more challenges, especially in terms of multi-modal problems.

6.2.2 Image Description

Image captioning is first regarded as a retrieval problem which aims to search captions from dataset for a given image [81, 82] and hence cannot provide accurate and proper descriptions for all images. To overcome this problem, methods like template filling [83] and paradigm for integrating convolutional neural network (CNN) and recurrent

*<https://github.com/bei21/img2poem>

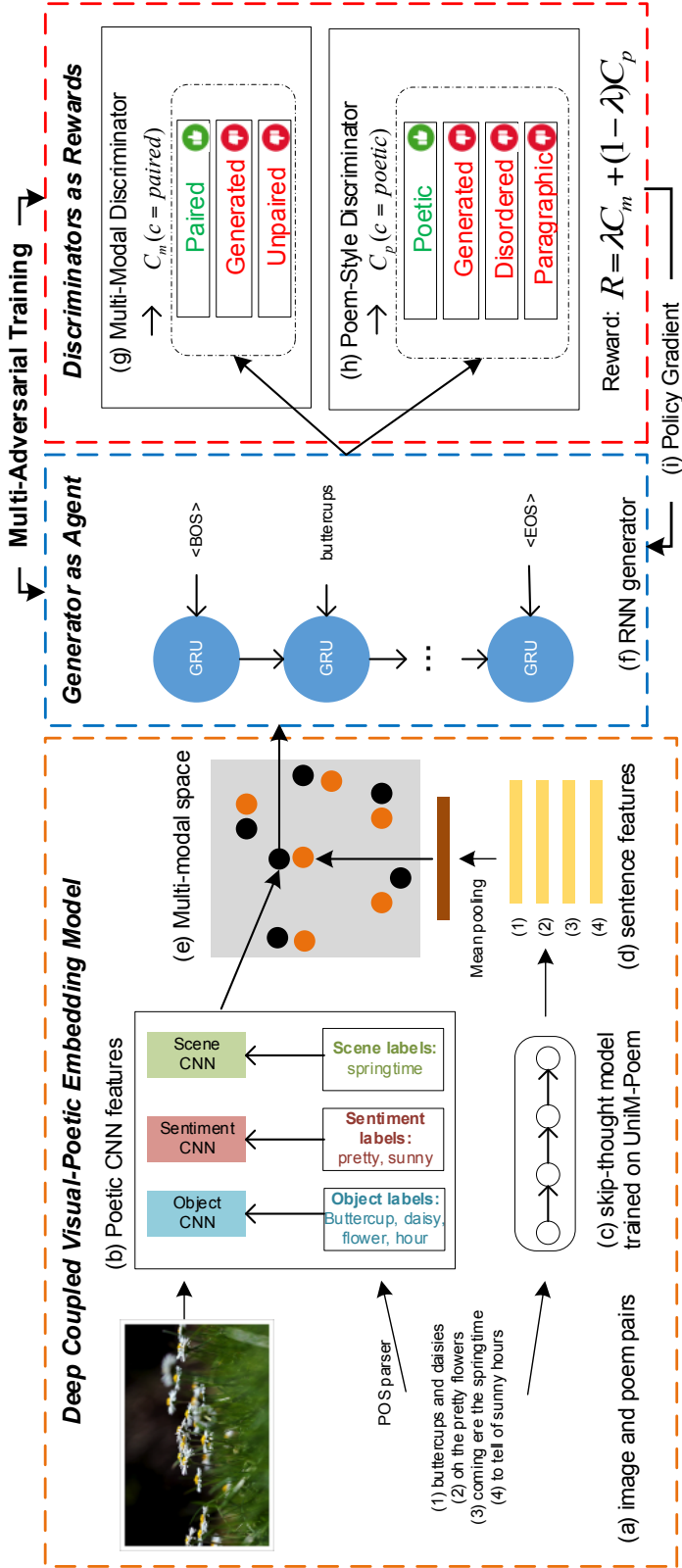


Figure 6.2. The framework of poetry generation with multi-adversarial training. A deep coupled visual-poetic model (e) is trained by human annotated image-poem pairs (a). The image features (b) are poetic multi-CNN features obtained by fine-tuning CNNs with the extracted poetic symbols (e.g., objects, scenes and sentiments) by a POS parser [2] from poems. The sentence features (d) of poems are extracted from a skip-thought model (c) trained on the largest public poem corpus (UniM-Poem). A RNN-based sentence generator (f) is trained as agent and two discriminators considering multi-modal (g) and poem-style (h) critics of a generated poem to a given image provide rewards to policy gradient (i). POS parser extracts Part-Of-Speech words from poems.

neural network (RNN) [84, 6, 85, 86] are proposed to generate readable human-level sentences. Recently, generative adversarial network (GAN) is applied to generate captions based on different problem settings [26, 77]. Similarly to image captioning, image paragraphing is going the similar way. Recent researches about image paragraphing mainly focus on region detection and hierarchical structure for generated sentences [7, 4, 78]. However, as we have addressed, image captioning and paragraphing aim to generate descriptive sentences to tell the facts in images, while poem generation is tackling an advanced form of linguistic form which requires poeticness and language style constraints.

6.3 Approach

In this research, we aim to generate poems from images so that the generated poems are relevant to input images and satisfy poeticness. For this purpose, we cast our problem in a multi-adversarial procedure [22] and further optimize it with a policy gradient [100, 101]. A CNN-RNN generative model acts as an *agent*. The parameters of this agent define a *policy* whose execution will decide which word to be picked as an *action*. When the agent has picked all words in a poem, it observes a *reward*. We define two discriminative networks to serve as rewards concerning whether the generated poem is a paired one with the input image and whether the generated poem is poetic. The goal of our poem generation model is to generate a sequence of words as a poem for an image to maximize the expected end reward. This policy-gradient method has shown significant effectiveness to many tasks without non-differentiable metrics [26, 102, 77].

As shown in Fig. (6.2), the framework consists of several parts: (1) a deep coupled visual-poetic embedding model to learn poetic representations from images, and (2) a multi-adversarial training procedure optimized by policy gradient. A RNN based generator serves as agent, and two discriminative networks provide rewards to the policy gradient.

6.3.1 Deep Coupled Visual-Poetic Embedding

The goal of visual-poetic embedding model [103, 104] is to learn an embedding space where points of different modality, e.g. images and sentences, can be projected to. In a similar way to image captioning problem, we assume that a pair of image and po-

em shares similar poetic semantics which makes the embedding space learnable. By embedding both images and poems to the same feature space, we can directly compute the relevance between a poem and an image by poetic vector representations of them. Moreover, the embedding feature can be further utilized to initialize a optimized representation of poetic clues for poem generation.

The structure of our deep coupled visual-poetic embedding model is shown in left part of Fig. (6.2). For the input of images, we leverage three deep convolutional neural networks (CNNs) concerning three aspects that indicate important poetic clues from images inspired from fine-grained problems [105], namely object ($\mathbf{v}_1$), scene ($\mathbf{v}_2$) and sentiment ($\mathbf{v}_3$), after conducting a prior user study about important factors for poem creation from images. We observed that concepts in poems are often imaginative and poetic while concepts in the classification datasets we use to train our CNN models are concrete and common. To narrow the semantic gap between the visual representation of images and the textual representation of poems, we propose to fine-tune these three networks with MultiM-Poem dataset. Specifically, frequent used keywords about object, sentiment and scenes in the poems are picked as label vocabulary, and then we build three multi-label datasets based on MultiM-Poem dataset for object, sentiment and scenes detection respectively. Once the multi-label datasets are built, we fine-tune the pre-trained CNN models on the three datasets independently, which is optimized by sigmoid cross entropy loss as shown in Eq. (6.1). After that, we adopt the D -dimension deep features for each aspect from the penultimate fully-connected layer of the CNN models, and get a concatenated N -dimension ($N = D \times 3$) feature vector $\mathbf{v} \in \mathbb{R}^N$ as input of visual-poetic embedding for each image:

$$loss = \frac{-1}{N} \sum_{n=1}^N (t_n \log p_n + (1 - t_n) \log(1 - p_n)), \quad (6.1)$$

$$\begin{aligned} \mathbf{v}_1 &= f_{\text{Object}}(I), & \mathbf{v}_2 &= f_{\text{Scene}}(I), \\ \mathbf{v}_3 &= f_{\text{Sentiment}}(I), & \mathbf{v} &= (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3). \end{aligned} \quad (6.2)$$

The output of visual-poetic embedding vector $\mathbf{x}$ is a K -dimension vector representing the image embedding with linear mapping from image features:

$$\mathbf{x} = \mathbf{W}_v \cdot \mathbf{v} + \mathbf{b}_v \in \mathbb{R}^K, \quad (6.3)$$

where $\mathbf{W}_v \in \mathbb{R}^{K \times N}$ is the image embedding matrix and $\mathbf{b}_v \in \mathbb{R}^K$ is the image bias vector. Meanwhile, representation feature vector of a poem is computed by skip-thought vectors[93], which is a popular unsupervised method to learn sentence embedding. We

train skip-thought model on unpaired UniM-Poem dataset and use it to provide a better sentence representation for poem sentences. Mean value of all sentences' combined skip-thought features (unidirectional and bidirectional) is denoted by $\mathbf{t} \in \mathbb{R}^M$ where M is the combined dimension. Similar to image embedding, the poem embedding is denoted as:

$$\mathbf{m} = \mathbf{W}_t \cdot \mathbf{t} + \mathbf{b}_t \in \mathbb{R}^K, \quad (6.4)$$

where $\mathbf{W}_t \in \mathbb{R}^{K \times M}$ for the poem embedding matrix and $\mathbf{b}_t \in \mathbb{R}^K$ for the poem bias vector. Finally, the image and poem are embedded together by minimizing a pairwise ranking loss with dot-product similarity:

$$\begin{aligned} L = & \sum_{\mathbf{x}} \sum_k \max(0, \alpha - \mathbf{x} \cdot \mathbf{m} + \mathbf{x} \cdot \mathbf{m}_k) \\ & + \sum_{\mathbf{m}} \sum_k \max(0, \alpha - \mathbf{m} \cdot \mathbf{x} + \mathbf{m} \cdot \mathbf{x}_k), \end{aligned} \quad (6.5)$$

where $\mathbf{m}_k$ is a contrastive (irrelevant unpaired) poem for image embedding $\mathbf{x}$, and vice-versa with $\mathbf{x}_k$. α denotes the contrastive margin. As a result, the model we trained will produce higher dot-product similarity between embedding features of image-poem pairs than similarity between randomly generated pairs.

6.3.2 Poem Generator as an Agent

A conventional CNN-RNN model for image captioning is used to serve as an agent. Instead of using hierarchical methods that are used recently in generating multiple sentences [7], we use a non-hierarchical recurrent model by treating the end-of-sentence token as a word in the vocabulary. The reason is that 1) poems often consist of fewer words compared with paragraphs; 2) there is lower consistent hierarchy between sentences of poems, which makes the hierarchy much more difficult to learn. We also conduct experiment with hierarchical recurrent language model as a baseline and we will show the result in the experiment part.

The generative model includes CNNs for image encoder and a RNN for poem decoder. The reason of using RNN instead of CNN for languages is that it can better encode the structure-dependent semantics of the long sentences which are widely observed in poems. In this research, we apply Gated Recurrent Units (GRUs) [106] for poem decoder for its simple structure and robustness to overfitting problem on less training data. We use image-embedding features learned by the deep coupled visual-poetic embedding model explained in Section 6.3.1 as input of image encoder. Suppose

θ is the parameters of the model. Traditionally, our target is to learn θ by maximizing the likelihood of the observed sentence $\mathbf{y} = y_{1:T} \in \mathbb{Y}^*$ where T is the maximum length of generated sentence (including $\langle \text{BOS} \rangle$ for start of sentence, $\langle \text{EOS} \rangle$ for end of sentence and line breaks) and $\mathbb{Y}^*$ denotes a space of all sequences of selected words.

Let $r(y_{1:t})$ denote the reward achieved at time t and $R(y_{1:T})$ is the cumulative reward, namely $R(y_{1:T}) = \sum_{t=1}^T r(y_{1:t})$. Let $p_\theta(y_t|y_{1:(t-1)})$ be a parametric conditional probability of selecting y_t at time step t given all the previous words $y_{1:(t-1)}$. p_θ is defined as a parametric function of policy θ . The reward of policy gradient in each batch can be computed as the sum over all sequences of valid actions as the expected future reward. To iterate over sequences of all possible actions is exponential, but we can further write it in expectation so that it can be approximated with an unbiased estimator:

$$J(\theta) = \sum_{y_{1:T} \in \mathbb{Y}^*} p_\theta(y_{1:T}) R(y_{1:T}) = \mathbb{E}_{y_{1:T} \sim p_\theta} \sum_{t=1}^T r(y_{1:t}). \quad (6.6)$$

We aim to maximize $J(\theta)$ by following its gradient:

$$\nabla_\theta J(\theta) = \mathbb{E}_{y_{1:T} \sim p_\theta} \left[\sum_{t=1}^T \nabla_\theta \log p_\theta(y_{1:t-1}) \right] \sum_{t=1}^T r(y_{1:t}). \quad (6.7)$$

In practice the expected gradient can be approximated using a Monte-Carlo sample by sequentially sample each y_t from the model distribution $p_\theta(y_t|y_{1:(t-1)})$ for t from 1 to T . As discussed in [102], a baseline b can be introduced to reduce the variance of the gradient estimate without changing the expected gradient. Thus, the expected gradient with a single sample is approximated as follow:

$$\nabla_\theta J(\theta) \approx \sum_{t=1}^T \nabla_\theta \log p_\theta(y_{1:t-1}) \sum_{t=1}^T (r(y_{1:t}) - b_t). \quad (6.8)$$

6.3.3 Discriminators as Rewards

A good poem for an image has to satisfy at least two criteria: the poem (1) is relevant to the image, and (2) has some sense of poectiness concerning proper length, poem's language style and consistence between sentences. Based on these two requirements, we propose two discriminative networks to guide the generated poem: multi-modal discriminator and poem-style discriminator. Deep discriminative networks have been shown of great effectiveness in text classification task [26, 77], especially for tasks that cannot establish good loss functions. In this research, both discriminators we propose have several classes including one positive class and several negative classes.

Multi-Modal Discriminator. Multi-modal discriminator (D_m) is used to guide the generated poem $\mathbf{y}$ related to corresponding image $\mathbf{x}$. It is trained to classify a poem into three classes: *paired* as positive examples, *unpaired* and *generated* as negative examples. *Paired* includes ground-truth paired poems for the input images. *Unpaired* poems are randomly sampled from unpaired poems of the input images in training data. D_m includes a multi-modal encoder, modality fusion layer and a classifier with softmax function:

$$\mathbf{c} = \text{GRU}_\rho(\mathbf{y}), \quad (6.9)$$

$$f = \tanh(W_x \cdot \mathbf{x} + b_x) \odot \tanh(W_c \cdot \mathbf{c} + b_c), \quad (6.10)$$

$$C_m = \text{softmax}(W_m \cdot f + b_m), \quad (6.11)$$

where ρ , W_x , b_x , W_c , b_c , W_m , b_m are parameters to be learned, $\odot$ is element-wise multiplication and C_m denotes the probabilities over three classes of the multi-modal discriminator. We utilize GRU-based sentence encoder for discriminator training. Eq. (6.11) provides way to generate the probability of $(\mathbf{x}, \mathbf{y})$ classified into each class as denoted by $C_m(c|\mathbf{x}, \mathbf{y})$ where $c \in \{\textit{paired}, \textit{unpaired}, \textit{generated}\}$.

Poem-Style Discriminator. In contrast with most poem generation researches that emphasize on meter, rhyme or other traditional poetic techniques, we focus on *free verse* which is an open form of poetry. Even though, we require our generated poems have the quality of *poeticness* as we define in Section 6.1. Without making specific templates or rules for poems, we propose a poem-style discriminator (D_p) to guide generated poems towards human written poems. In D_p , generated poems will be classified into four classes: *poetic*, *disordered*, *paragraphic* and *generated*.

Class *poetic* is addressed as positive example of poems that satisfy poeticness. The other three classes are all regarded as negative examples. Class *disordered* concerns about the inner structure and coherence between sentences of poems and *paragraphic* class uses paragraph sentences as negative examples. In D_p , we use UniM-Poem as positive *poetic* samples. To construct disordered poems, we first construct a poem sentence pool by splitting all poems in UniM-Poem. Examples of class *disordered* are poems that we reconstruct by sentences randomly picked up with a reasonable line numbers from poem sentence pool. Paragraph dataset provided by [7] is used as *paragraphic* examples.

A completed generated poem $\mathbf{y}$ is encoded by GRU and parsed to a fully connected layer, and the probability of falling into four classes is computed by a softmax function.

Formula of this procedure is as follow:

$$C_p = \text{softmax}(W_p \cdot \text{GRU}_\eta(\mathbf{y}) + b_p), \quad (6.12)$$

where η , W_p , b_p are parameters to be learned. The probability of classifying generated poem $\mathbf{y}$ to a class c is formulated as $C_p(c|\mathbf{y})$ where $c \in \{\text{poetic}, \text{disordered}, \text{paragraphic}, \text{generated}\}$.

Reward Function. We define the reward function for policy gradient as a linear combination of probability of classifying generated poem $\mathbf{y}$ for an input image $\mathbf{x}$ to the positive class (*paired* for multi-modal discriminator D_m and *poetic* for poem-style discriminator D_p) weighted by tradeoff parameter λ :

$$R(\mathbf{y}|\cdot) = \lambda C_m(c = \text{paired}|\mathbf{x}, \mathbf{y}) + (1 - \lambda) C_p(c = \text{poetic}|\mathbf{y}). \quad (6.13)$$

6.3.4 Multi-Adversarial Training

Adversarial training is a minimax game between a generator G and a discriminator D with value function $V(G, D)$:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]. \quad (6.14)$$

We propose to use multiple discriminators by reformulating G 's objective as:

$$\min_G F(V(D_1, G), \dots, V(D_n, G)), \quad (6.15)$$

where we have $n = 2$, and F indicates linear combination of discriminators as shown in Eq. (6.13).

The generator aims to generate poems that have higher rewards for both discriminators so that they can fool the discriminators while the discriminators are trained to distinguish the generated poems from *paired* and *poetic* poems. The probabilities of classifying generated poem into positive classes in both discriminators are used as rewards to policy gradient as explained above.

Multiple discriminators (two in this work) are trained by providing positive examples from the real data (paired poems in D_m and poem corpus in D_p) and negative examples from poems generated from the generator as well as other negative forms of real data (unpaired poems in D_m , paragraphs and disordered poems in D_p). Meanwhile, by employing a policy gradient and Monte Carlo sampling, the generator is updated based on the expected rewards from multiple discriminators. Since we have two discriminators, we apply a multi-adversarial training method that will train two discriminators in a parallel way.

	spreading his arms dancing the floor through the light Towards a new dawn		burning base of sky half the hemisphere high a fire hour crept upon by night
	the morning sunrise so beautiful to behold glowing breath of light		back on its golden hinges the gate of memory swings and my heart goes into the garden and walks with the olden things

Image and poem pair dataset (MultiM-Poem)

in crescent form a vasty crescent nigh two leagues across from horn to horn the lesser ships within the great without they did bestride as 't were and make a township on the narrow seas	this realm of rain grey sky and cloud it's quite and peaceful safe allowed
his little hands when flowers were seen were held for the bluebell as he was carried oer the green	in a brown gloom the moats gleam slender the sweet wife stands her lips are red her eyes dream kisses are warm on her hands

Poem Corpus Dataset (UniM-Poem)

Figure 6.3. Examples in two datasets: UniM-Poem and MultiM-Poem.

6.4 Experiments

6.4.1 Datasets

To facilitate the research of poetry generation from images, we collected two poem datasets, in which one consists of image and poem pairs, namely Multi-Modal Poem dataset (MultiM-Poem), and the other is a large poem corpus, namely Uni-Modal Poem dataset (UniM-Poem). By using the embedding model we have trained, the image and poem pairs are extended by adding the nearest three neighbor poems from the poem corpus without redundancy, and an extended image and poem pair dataset is constructed and denoted as MultiM-Poem (Ex). The detailed information about these datasets is listed in Table 6.1. Examples of the two collected datasets can be seen in Figure 6.3.

Name	#Poem	#Line/poem	#Word/line
MultiM-Poem	8,292	7.2	5.7
UniM-Poem	93,265	5.7	6.2
MultiM-Poem (Ex)	26,161	5.4	5.9

Table 6.1. Detailed information about the three datasets. The first two datasets are collected by ourselves and the third one is extended by our embedding model.

For MultiM-Poem dataset, we first crawled 34,847 image-poem pairs in Flickr from groups that aim to use images illustrating poems written by human. Five human assessors majoring in English literature were further asked to evaluate these poems as relevant or irrelevant by judging whether the image can exactly inspire the poem in a pair by considering the associations of objects, sentiments and scenes. We filtered out pairs labeled as irrelevant and kept the remaining 8,292 pairs to construct the MultiM-Poem dataset.

UniM-Poem is crawled from several public online poetry websites, such as Poetry Foundation[†], PoetrySoup[‡], best-poem.net and poets.org. To achieve robust model training, a poem pre-processing procedure is conducted to filter out those poems with too many lines (> 10) or too fewer lines (< 3). We also remove poems with strange characters, poems in languages other than English and duplicate poems.

6.4.2 Compared Methods

To investigate the effectiveness of the proposed methods, we compare with four baseline models with different settings. The models of show-and-tell [6] and SeqGan [77] are selected due to their state-of-art results in image captioning. A competitive image paragrphing model is selected, as its strong capability for modeling diverse image content. Note that all the methods use MultiM-Poem (Ex) as the training dataset, and can generate multiple lines as poems. The detailed experiment settings are shown as follows:

Show and tell (1CNN): CNN-RNN model trained with only object CNN by VGG-16 .

Show and tell (3CNNs): CNN-RNN model trained with three CNN features by

[†]<https://www.poetryfoundation.org/>

[‡]<https://www.poetrysoup.com/>

VGG-16.

SeqGAN: CNN-RNN model optimized with a discriminator to tell from generated poems and ground-truth poems. We use RNN for discriminator for fair comparison.

Regions-Hierarchical: Hierarchical paragraph generation model based on [7]. To better align with poem distribution, we restrict the maximum lines to be 10 and each line has up to 10 words in the experiment.

Our Model: To demonstrate the effectiveness of the two discriminators, we train our model (Image to Poem with GAN, I2P-GAN) in four settings: pretrained model without discriminators (**I2P-GAN w/o discriminator**), with multi-modal discriminator only (**I2P-GAN w/ D_m**), with poem-style discriminator only (**I2P-GAN w/ D_p**) and with both discriminators (**I2P-GAN**).

6.4.3 Automatic Evaluation Metrics

Evaluation of poems is generally a difficult task and there are no established metrics in existing works, not to mention the new task of generating poems from images. To better address the performance of the generated poems, we propose to evaluate them in both automatic and manual way.

We propose to employ three metrics for automatic evaluation, e.g., BLEU, novelty and relevance. An overall score is computed by the three metrics after normalization.

BLEU. We use Bilingual Evaluation Understudy (BLEU) [107] score-based evaluation to examine how likely the generated poems can approximate towards the ground-truth ones following image captioning and paragraphing. It is also used in some poem generation works [89]. For each image, we only use the human written poems as ground-truth poems.

Novelty. By introducing discriminator D_p , the generator is supposed to introduce words or phrases from UniM-Poem dataset and results in words or phrases that are not very frequent in MultiM-Poem (Ex) dataset. We use *novelty* as proposed by [10] to measure the number of infrequent words or phrases observed in the generated poems. Two scales of N-gram are explored, e.g. bigram and trigram, as **Novelty-2** and **Novelty-3**. We first rank the n-grams that occur in the training dataset of MultiM-Poem (Ex) and take the top 2,000 as frequent ones. Novelty is computed as the proportion of n-grams that occur in training dataset except the frequent ones in the generated poem.

Relevance. Different from poem generation researches that have no or weak constraints to poem contents, we consider relevance of the generated poem to the given

image as an important measurement in this research. However, unlike captions that concern more about facts about images, different poems can be relevant to the same image from various aspects. Thus, instead of computing relevance between generated poem and ground-truth poems, we define relevance between a poem and an image using our learned deep coupled visual-poetic embedding (VPE) model. After mapping the image and the poem to the same space through VPE, linearly scaled cosine similarity (0-1) is used to measure their relevance.

Overall. We compute an overall score based on the above three metrics. For each value a_i in all values of one metric $\mathbf{a}$, we first linearly normalize it with following method:

$$a_i' = \frac{a_i - \min(a)}{\max(a) - \min(a)}. \quad (6.16)$$

After that, we get average values for BLEU (e.g. BLEU-1, BLEU-2 and BLEU-3) and novelty (e.g. Novelty-2 and Novelty-3). A final score is computed by averaging the normalized values, to ensure equal contribution of different metrics.

However, in such an open-ended task, there are no particularly suitable metrics that can perfectly evaluate the performance of generated poems. The automatic metrics we use can be used as a guidance to some extent. To better illustrate the performance of poems from human perception, we further conduct extensive user studies in the follows.

6.4.4 Human Evaluation

We conducted human evaluation in Amazon Mechanical Turk. In particular, three types of tasks are assigned:

Task1: to explore the effectiveness of our deep coupled visual-poetic embedding model, annotators were requested to give a 0-10 scale score to a poem given an image considering their relevance in case of content, emotion and scene.

Task2: this task aims to compare the generated poems by different methods (four baseline methods and our four model settings) for one image on different aspects. Given an image, the annotators were asked to give ratings to a poem on a 0-10 scale with respect to four criteria: relevance (to the image), coherence (whether the poem is coherent across lines), imaginativeness (how much imaginative and creative the poem is for the given image) and overall impression.

Task3: Turing test was conducted by asking annotators to select human written poem from mixed human written and generated poems. Note that Turing test was

implemented in two settings, i.e., with and without images as references.

For each task, we have randomly picked up 1K images and each task is assigned to three assessors. As poem is a form of literature, we also ask 30 annotators whose majors are related to English literature (among which ten annotations are English natives) as *expert users* to do the Turing test.

6.4.5 Training Details

In the deep coupled visual-poetic embedding model, we use $D = 4,096$ -dimension “fc7” features for each CNN. Object features are extracted from VGG-16 [71] trained on ImageNet [108], scene features from Place205-VGGNet model [109], and sentiment features from sentiment model[63].

To better extract visual feature for poetic symbols, we first get nouns, verbs and subjective adjectives with at least five frequency in UniM-Poem dataset. Then we manually picked adjectives and verbs for sentiment (including 328 labels), nouns for object (including 604 labels) and scenes (including 125 labels). As for poem features, we extract a combined skip-thought vector with $M = 2,048$ -dimension (in which each 1,024-dimension represents for uni-direction and bi-direction, respectively) for each sentence, and finally we get poem features by mean pooling. And the margin α is set to 0.2 based on empirical experiments in [104]. We randomly select 127 poems as unpaired poems for an image and used them as contrastive poems ($\mathbf{m}_k$ and $\mathbf{x}_k$ in Eq. (6.5)), and we re-sample them in each epoch. Before adversarial training, we pre-train a generator based on image captioning method [6] which can provide a better policy initialization for generator. We empirically set the tradeoff parameter $\lambda = 0.8$ by conducting a comparable observation on automatic evaluation results from 0.1 to 0.9.

6.4.6 Evaluations

Retrieved Poems. We compare three kinds of poems considering their relevance to images: ground-truth poems, poems retrieved with VPE and image features before fine-tuning (VPE w/o FT), and poems retrieved with VPE and fine-tuned image features (VPE w/ FT). Table 6.2 shows a comparison on a scale of 0-10 (0 means irrelevant and 10 means the most relevant). We can see that by using the proposed visual-poetic embedding model, the retrieved poems can achieve a relevance score above the average score (i.e., the score of five). And image features fine-tuned with poetic symbols

	Ground-Truth	VPE w/o FT	VPE w/ FT
Relevance	7.22	5.82	6.32

Table 6.2. Average score of relevance to images for three types of human written poems on 0-10 scale (0-irrelevant, 10-relevant). One-way ANOVA revealed that evaluation on these poems is statistically significant ($F(2,9) = 130.58, p < 1e - 10$).

can improve the relevance significantly.

Generated Poems. Table 6.6 exhibits the automatic evaluation results of the proposed model with four settings, as well as the four baselines proposed in previous works. Comparing results of caption model with one CNN and three CNNs, we can see that multi-CNN can actually help to generate poems that are more relevant to images. Regions-Hierarchical model emphasizes more on the topic coherence between sentences while many human written poems will cover several topics or use different symbols for one topic. SeqGAN shows the advantage of applying adversarial training for poem generation compared with only caption models with only CNN-RNN while lacking of generating novel concepts in poems. Better performance of our pre-trained model with VPE than caption model demonstrates the effectiveness of VPE in extracting poetic features from images for better poem generation. We can see that our three models outperform in most of the metrics with each one performs better at one aspect. The model with only multi-modal discriminator (**I2P-GAN w/ D_m**) will guide the model to generate poems towards ground-truth poems, thus it results in the highest BLEU scores that emphasize the similarity of n-grams in a translative way. Poem-style discriminator (D_p) is designed to guide the generated poem to be more poetic in language style, and the highest novelty score of **I2P-GAN w/ D_m** shows that D_p helps to provide more novel and imaginative words to the generated poem. Overall, **I2P-GAN** combines the advantages of both discriminators with a rational intermediate score regarding BLEU and novelty while still outperforms compared with other generation models. Moreover, our model with both discriminators can generate poems that have highest relevance on our embedding relevance metric.

Comparison of human evaluation results are shown in Table 6.3. Different from automatic evaluation results where Regions-Hierarchical performs not well, it gets a slightly better result than caption model for the reason that sentences all about the same topic tend to gain better impressions from users. Our three models outperform the other four baseline methods on all metrics. Two discriminators promote human-

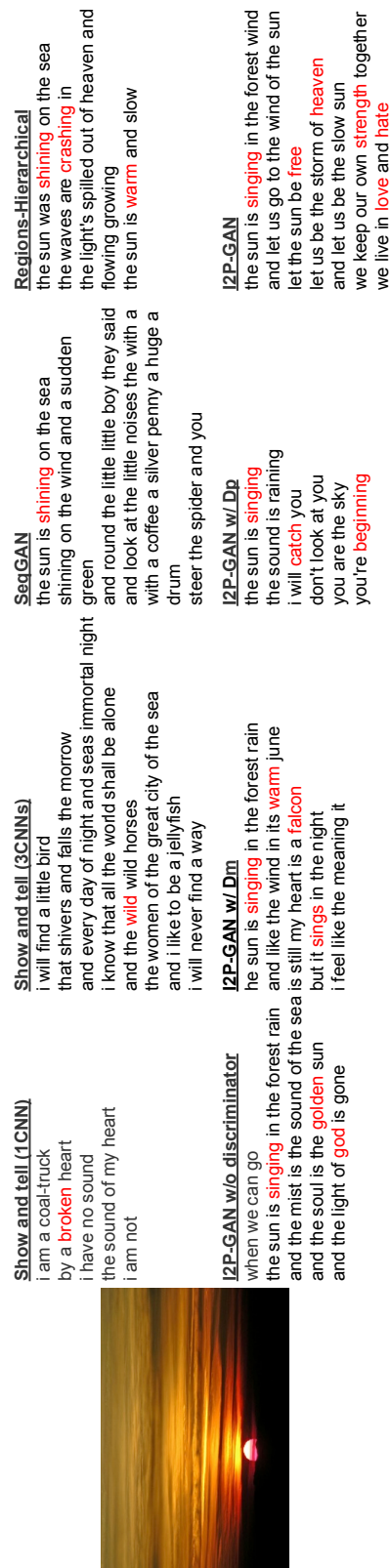


Figure 6.4. Example of poems generated by eight methods for an image. Words in read indicate poeticness.

6. Poetry Generation from Image

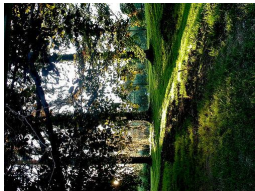
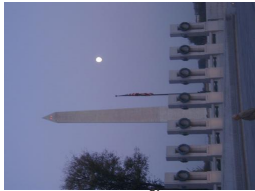
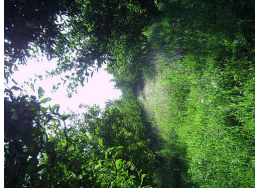
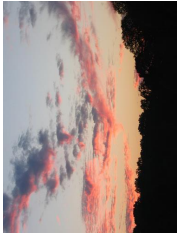
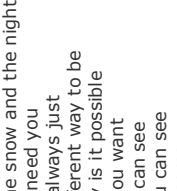
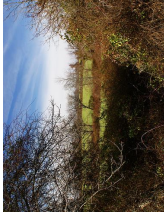
	the sun rays struck my face warm tingles to my fingertips the light showed me a path i should walk down i spoke and the whispers of the breeze told me to close my eyes i lost my way in a paradise		i have been a great city spinning and shout the sound of the road washed away the mountain passes through the streets are gone the silence is raining it sits still in silence glint its own		the sun is a beautiful thing in silence is drawn between the trees only the beginning of light
	i will arise and go by the sea gate i watch it fly and let me lie in the dark green valleys and let me sing to the sun i know you are not beautiful enough to me and you know that you are so much you are so much you can see you love you will never fly if you are always		when the sun shines through the snow and the night is somebody really need you if only can be really always just just as if you can be a different way to be walking through the sky is it possible you know that is you want to be pure as you can see the light leaves you can see the cherry blossoms		
	and now i am tired of my own let me be the freshening blue haunted through the sky bare and cold water warm blue air shimmering brightly never arrives it seems to say		the sun is shining the wind moves naked trees you dance		i have seen the wind blows my heart filled with air my eyes are bleared with stinging i am a woman is a reminder of love that's just just one thing

Figure 6.5. Example of poems generated by our approach I2P-GAN.

Method	Relevance	Novelty-2	Novelty-3	BLEU-1	BLEU-2	BLEU-3	Overall
Show and Tell (1CNN)[6]	1.79	43.66	76.76	11.88	3.35	0.76	14.40
Show and Tell (3CNNs)[6]	1.91	48.09	81.37	12.64	3.34	0.8	34.34
SeqGAN[77]	2.03	47.52	82.32	13.40	3.72	0.76	44.95
Regions-Hierarchical[7]	1.81	46.75	79.90	11.64	2.5	0.67	8.01
I2P-GAN w/o discriminator	1.94	45.25	80.13	13.35	3.69	0.88	41.86
I2P-GAN w/ D_m	2.07	43.37	78.98	15.15	4.13	1.02	63.00
I2P-GAN w/ D_p	1.90	60.66	89.74	12.91	3.05	0.72	51.35
I2P-GAN	2.25	54.32	85.37	14.25	3.84	0.94	77.23

Figure 6.6. Automatic evaluation. Note that BLEU scores are computed in comparison with human-annotated ground-truth poems (one poem for one image). Overall score is computed as an average of three metrics after normalization (Eq. (6.16)). All scores are reported as percentage (%).

Method	Rel	Coh	Imag	Overall
Show and Tell (1CNN)[6]	6.31	6.52	6.57	6.67
Show and Tell (3CNNs)[6]	6.41	6.59	6.63	6.75
SeqGAN[77]	6.13	6.43	6.50	6.63
Regions-Hierarchical[7]	6.35	6.54	6.63	6.78
I2P-GAN w/o discriminator	6.44	6.64	6.77	6.85
I2P-GAN w/ D_m	6.59	6.83	6.94	7.06
I2P-GAN w/ D_p	6.53	6.75	6.80	6.93
I2P-GAN	6.83	6.95	7.05	7.18
Ground-Truth	7.10	7.26	7.23	7.37

Table 6.3. Human evaluation results of six methods on four criteria: relevance (Rel), coherence (Coh), imaginativeness (Imag) and Overall. All criteria are evaluated on 0-10 scale (0-bad, 10-good).

Data	Users	Ground-Truth	Generated
Poem w/ Image	AMT	0.51	0.49
	Expert	0.60	0.40
Poem w/o Image	AMT	0.55	0.45
	Expert	0.57	0.43

Table 6.4. Accuracy of Turing test on AMT users and expert users on poems with and without images.

level comprehension towards poems compared with pre-trained model. The model with two discriminators has generated better poems from images in terms of relevance, coherence and imaginativeness. Fig. (6.4) shows one example of poems generated with three baselines and our methods for a given image. More examples generated by our approach can be referred in the supplementary material.

Turing Test. For the Turing test of annotators in AMT, we have hired 548 workers with 10.9 tasks for each one on average. For experts, 15 people were asked to judge human written poems with images and another 15 annotators were asked to do test with only poems. Each one is assigned with 20 images and in total we have 600 tasks conducted by expert users. Table 6.4 shows the probability of different poems being

selected as human-written poems for an given image. As we can see, the generated poems have caused a competitive confusion to both ordinary annotators and experts though experts can figure out the accurate one better than ordinary people. One interesting observation comes from that experts are better at figuring out correct ones with images while AMT workers do better with only poems.

6.5 Conclusion and Discussion

As the frontal work of poetry (English free verse) generation from images, we propose a novel approach to model the problem by incorporating deep coupled visual-poetic embedding model and RNN based adversarial training with multi-discriminators as rewards for policy gradient. Furthermore, we introduce the first image and poem pair dataset (MultiM-Poem) and a large poem corpus (UniM-Poem) to enhance researches on poem generation, especially from images. Extensive experiments demonstrated that our embedding model can approximately learn a rational visual-poetic embedding space. Objective and subjective evaluation results demonstrated the effectiveness of our poem generation model.

Generating poems from images is brand new topic that has attracted extensive interest from researchers. There are two main challenges facing with this task. The first is how to make machines better learn poetic clues from images. As we have discussed, the poetic inspiration from images might need our past experiences which a machine does not have. Something like a knowledge graph might be needed to simulate this kind of experiences of human. The second is how to make expert-level poems like famous poets. This is also a challenge in many natural language processing tasks. A key problem might be the lack of a perfect evaluation metric to make the machine aware of good or bad poems. This is still an open question that will attracts more discussion and researches.

CONCLUSION AND FUTURE WORK

7.1 Conclusion

In this work, we proposed a very broad framework of high-level cognitive understanding of images towards language. To this end, two common types of problems were discussed in this framework, including search-based tasks and generation-based tasks. Although our current research achievements cannot take into account all aspects of this big problem, we have tackled the following particular sub-problems.

Event Summarization with Images. We propose to achieve cognition-aware summarization of images presenting events from the perspective of viewers rather than image takers. To improve the perceptual quality of an image set, we analyze relationship of events in a hierarchical way and define three types of neighbor events that could be easily recognized as the target event. The reasons for highly possible misrecognitions are analyzed, and three criteria are raised to measure the degree to prevent from them. We propose a greedy algorithm to generate image sets by maximizing the objective function that combines the three criteria.

Learning Subjective Adjectives from Images. We propose to solve the problem of estimating relevance of images to subjective adjective queries by first learning discriminative features of subjective adjective from images with unsupervised deep convolution auto-encoders and then learn to measure the relevance. We propose pairwise stacked convolutional auto-encoders to find discriminative features that can represent

differences between relevant images and irrelevant images.

Visual Storytelling. We propose to introduce emotion as an important factor to generate story from image stream. We incorporate an emotion prediction model which is able to predict diverse and coherent emotions and a coupled-RNN story generator with image stream and emotion as input, in which two discriminators and a similarity function provide rewards to measure image relevance, emotion relevance and story style.

Poetry Generation from Image. We propose a novel approach to model the problem of poetry generation from image by incorporating deep coupled visual-poetic embedding model and RNN-based adversarial training with multi-discriminators as rewards for policy gradient. Discriminator that addresses the relevance of generated poem and input image and discriminator that deals with the poem language style of generated poem are explored to improve the poem generator. We also introduce the first image and poem pair dataset and a large poem corpus to enhance researches of poem generation, especially from images.

Extensive experiments have been conducted to these tasks and the results have demonstrated the effectiveness of our proposed approach compared with baseline methods.

7.2 Future Work

The proposed research framework in this thesis is intended to inspire more interests and attention in social informatics research, especially image and language related researches. As another goal, we aim to improve machine’s understanding of images from the perspective of human cognition. Accordingly, we have the following future plan.

Multi-Modal Embedding. We feel and perceive the world with multiple ways. Image and language are the two among them. Other types include but only limited to are: sound, video, touch and taste. As long as the machine is able to stimulate each way of sense, we could learn the embedding between or among them. Current researches allow us to represent image, sound and language very well. How to better learn the embedding space where we can directly match or compare different modality is a very interesting direction.

Application Diversity. In this research, we tackle four tasks that try to bridge the image and language in a high-level cognitive way. More research topics could be

involved, such as visual dialog that aims to automatically generate conversion to an image or several images and visual question and answer that tries to automatically generate answers to an image and related question. A very exciting vision is that machines could look at the world as we see, and understand the world as we understand.

ACKNOWLEDGEMENTS

I would like to express my gratitude to all those who helped me during the writing of this thesis.

My deepest gratitude goes first and foremost to my supervisors, Professor Katsumi Tanaka and Professor Masatoshi Yoshikawa, for their constant encouragement and guidance. In addition, I would like to thank senior lecturer Makoto P. Kato for his valuable advices and support during my master and doctoral research life. I would like to thank my other academic advisors, Dr. Tao Mei and Dr. Jianlong Fu, for their valuable suggestions and comments on my research. Thanks to Professor Takayuki Kanda and Professor Shinsuke Mori for being a member of the thesis committee.

I would like to thank all our lab faculties for their help in every possible way. During the lab seminars, Associate Professor Adam Jatowt, Associate Professor Hiroaki Ohshima, Assistant Professor Takehiro Yamamoto, Mr. Osami Kagawa, Associate Professor Qiang Ma, Associate Professor Yasuhito Asano, Assistant Professor Toshiyuki Shimizu, made valuable comments on my research, which inspired me a lot of new ideas. Also, the lab secretary Ms. Rika Ikebe helped me a lot on daily business and life.

Special thanks should go to my friends who gave me their help and time in listening to me, helping me work out my problems and accompanying me to make life much interesting and better during the difficult course of the thesis.

Last my thanks would go to my beloved family for their loving considerations and great confidence in me all through these years.

Bei LIU, September 2018

REFERENCES

- [1] Jing Wang, Jianlong Fu, Jinhui Tang, Zechao Li, and Tao Mei. Show, reward and tell: Automatic generation of narrative paragraph from photo stream by adversarial training. *AAAI*, 2018.
- [2] Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *HLT-NAACL*, pages 173–180, 2003.
- [3] Steven Pinker. *The language instinct: How the mind creates language*. Penguin UK, 2003.
- [4] Yu Liu, Jianlong Fu, Tao Mei, and Chang Wen Chen. Let your photos talk: Generating narrative paragraph for photo stream via bidirectional attention recurrent neural networks. In *AAAI*, 2017.
- [5] Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh K Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C Platt, et al. From captions to visual concepts and back. In *CVPR*, pages 1473–1482, 2015.
- [6] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *CVPR*, pages 3156–3164, 2015.
- [7] Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. A hierarchical approach for generating descriptive image paragraphs. *CVPR*, pages 3337–3345, 2017.
- [8] Ting-Hao Kenneth Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet

- Kohli, Dhruv Batra, et al. Visual storytelling. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1233–1239, 2016.
- [9] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433, 2015.
- [10] Zhen Xu, Bingquan Liu, Baoxun Wang, SUN Chengjie, Xiaolong Wang, Zhuoran Wang, and Chao Qi. Neural response generation via gan with an approximate embedding layer. In *EMNLP*, pages 628–637, 2017.
- [11] Shuang Ma, Jianlong Fu, Chang Wen Chen, and Tao Mei. Da-gan: Instance-level image translation by deep attention generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5657–5666, 2018.
- [12] Janyce Wiebe. Learning subjective adjectives from corpora. In *AAAI/IAAI*, pages 735–740, 2000.
- [13] Jaime Carbonell and Jade Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proc. 21st Ann. Int. Conf. on Research and Development in Information Retrieval, SIGIR '98*, pages 335–336, 1998.
- [14] Jürgen Schmidhuber, Martin Eldracher, and Bernhard Foltin. Semilinear predictability minimization produces well-known feature detectors. *Neural Computation*, 8(4):773–786, 1996.
- [15] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.
- [16] Fu Jie Huang, Y-Lan Boureau, Yann LeCun, et al. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *2007 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2007.

- [17] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.
- [18] Jonathan Masci, Ueli Meier, Dan Cireşan, and Jürgen Schmidhuber. Stacked convolutional auto-encoders for hierarchical feature extraction. In *International Conference on Artificial Neural Networks*, pages 52–59. Springer, 2011.
- [19] Yoshua Bengio, Pascal Lamblin, Dan Popovici, Hugo Larochelle, et al. Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, 19:153, 2007.
- [20] Hugo Larochelle, Dumitru Erhan, Aaron Courville, James Bergstra, and Yoshua Bengio. An empirical evaluation of deep architectures on problems with many factors of variation. In *Proceedings of the 24th international conference on Machine learning*, pages 473–480. ACM, 2007.
- [21] Xiaoyuan Yi, Ruoyu Li, and Maosong Sun. Generating chinese classical poems with rnn encoder-decoder. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, pages 211–223. 2017.
- [22] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.
- [23] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint*, 2017.
- [24] Zili Yi, Hao (Richard) Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *ICCV*, pages 2868–2876, 2017.
- [25] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

- [26] Tseng-Hung Chen, Yuan-Hong Liao, Ching-Yao Chuang, Wan-Ting Hsu, Jian-long Fu, and Min Sun. Show, adapt and tell: Adversarial training of cross-domain image captioner. *ICCV*, pages 521–530, 2017.
- [27] Michael L Littman, Andrew W Moore, et al. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4(11, 28):237–285, 1996.
- [28] Richard S Sutton and Andrew G Barto. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.
- [29] Seyed Sajad Mousavi, Michael Schukat, and Enda Howley. Deep reinforcement learning: an overview. In *Proceedings of SAI Intelligent Systems Conference*, pages 426–440. Springer, 2016.
- [30] Xiao Liu, Tian Xia, Jiang Wang, Yi Yang, Feng Zhou, and Yuanqing Lin. Fully convolutional attention networks for fine-grained recognition. *arXiv preprint arXiv:1603.06765*, 2016.
- [31] Stefan Mathe, Aleksis Pirinen, and Cristian Sminchisescu. Reinforcement learning for visual object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2894–2902, 2016.
- [32] Zhou Ren, Xiaoyu Wang, Ning Zhang, Xutao Lv, and Li-Jia Li. Deep reinforcement learning-based image captioning with embedding reward. *arXiv preprint arXiv:1704.03899*, 2017.
- [33] Ling Chen and Abhishek Roy. Event detection from flickr data through wavelet-based spatial analysis. In *Proc. 18th Conf. on Information and Knowledge Management, CIKM ’09*, pages 523–532. ACM, 2009.
- [34] Pinaki Sinha, Sharad Mehrotra, and Ramesh Jain. Summarization of personal photologs using multidimensional content and context. In *Proc. 1st Int. Conf. on Multimedia Retrieval, ICMR ’11*, pages 4:1–4:8. ACM, 2011.
- [35] Pinaki Sinha and Ramesh Jain. Extractive summarization of personal photos from life events. In *Proc. 2011 Int. Conf. on Multimedia and Expo, ICME ’11*, pages 1–6. IEEE, 2011.
- [36] Jürgen Bohnemeyer and Eric Pederson. *Event representation in language and cognition*. Cambridge University Press, 2011.

- [37] Apoorv Agarwal and Owen Rambow. Automatic detection and classification of social events. In *Proc. 2010 Conf. on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 1024–1034, 2010.
- [38] Duc-Duy Nguyen, Minh-Son Dao, and Truc-Vien T Nguyen. Natural language processing for social event classification. In *Proc. 6th Int. Conf. on Knowledge and Systems Engineering*, pages 79–91, 2015.
- [39] Kevin Tang, Li Fei-Fei, and Daphne Koller. Learning latent temporal structure for complex event detection. In *Proc. Conf. on Computer Vision and Pattern Recognition*, pages 1250–1257. IEEE, 2012.
- [40] Zhigang Ma, Yi Yang, Zhongwen Xu, Shuicheng Yan, Nicu Sebe, and Alexander G Hauptmann. Complex event detection via multi-source video attributes. In *Proc. Conf. on Computer Vision and Pattern Recognition*, pages 2627–2633. IEEE, 2013.
- [41] Markus Brenner and Ebroul Izquierdo. Social event detection and retrieval in collaborative photo collections. In *Proc. 2nd Int. Conf. on Multimedia Retrieval*, ICMR '12, page 21. ACM, 2012.
- [42] John C Platt, Mary Czerwinski, and Brent A Field. Phototoc: Automatic clustering for browsing personal photographs. In *Proc. 2003 Joint Conf. on Information, Communications and Signal Processing and 4th Pacific Rim Conf. on Multimedia*, volume 1, pages 6–10 Vol.1, 2003.
- [43] Mor Naaman, Yee Jiun Song, Andreas Paepcke, and Hector Garcia-Molina. Automatic organization for digital photographs with geographic coordinates. In *Proc. 4th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '04, pages 53–62. ACM, 2004.
- [44] Pinaki Sinha. Summarization of archived and shared personal photo collections. In *Proc. 20th Int. Conf. Companion on World Wide Web*, WWW '11, pages 421–426. ACM, 2011.
- [45] Gunhee Kim, Leonid Sigal, and Eric P Xing. Joint summarization of large-scale collections of web images and videos for storyline reconstruction. In *Proc. 2014 Conf. on Computer Vision and Pattern Recognition*, CVPR '14, pages 4225–4232. IEEE, 2014.

- [46] Matthew Cooper, Jonathan Foote, Andreas Girgensohn, and Lynn Wilcox. Temporal event clustering for digital photo collections. *Trans. on Multimedia Computing, Communications, and Applications*, 1(3):269–288, 2005.
- [47] Alexander C Loui and Andreas Savakis. Automated event clustering and quality screening of consumer pictures for digital albuming. *Trans. Multimedia*, 5(3):390–402, 2003.
- [48] Naveed Imran, Jingen Liu, Jiebo Luo, and Mubarak Shah. Event recognition from photo collections via pagerank. In *Proc. 17th Int. Conf. on Multimedia*, MM ’09, pages 621–624. ACM, 2009.
- [49] Chung-Lin Wen et al. *Event-centric Twitter photo summarization*. PhD thesis, Massachusetts Institute of Technology, 2014.
- [50] Jeffrey M Zacks and Barbara Tversky. Event structure in perception and conception. *Psychological bulletin*, 127(1):3–21, 2001.
- [51] Miles Osborne and Mark Dredze. Facebook, twitter and google plus for breaking news: Is there a winner? In *Proc. 8th Int. Conf. on Weblogs and Social Media*, ICWSM ’14. AAAI Press, 2014.
- [52] Takeshi Sakaki, Masahide Okazaki, and Yoshikazu Matsuo. Tweet analysis for real-time event detection and earthquake reporting system development. *Trans. on Knowledge and Data Engineering*, 25(4):919–931, 2013.
- [53] Jeffrey M Zacks, Nicole K Speer, Khen M Swallow, Todd S Braver, and Jeremy R Reynolds. Event perception: a mind-brain perspective. *Psychological bulletin*, 133(2):273–293, 2007.
- [54] Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. Diversifying search results. In *Proc. 2nd Int. Conf. on Web Search and Data Mining*, WSDM ’09, pages 5–14. ACM, 2009.
- [55] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When is “nearest neighbor” meaningful? In *Proc. 7th Int. Conf. on Database Theory*, ICDT ’99, pages 217–235. Springer-Verlag, 1999.

- [56] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functionsi. *Mathematical Programming*, 14(1):265–294, 1978.
- [57] Gerard Salton and Michael J. McGill. *Introduction to modern information retrieval*. McGraw-Hill, Inc., 1986.
- [58] Carina Silberer, Vittorio Ferrari, and Mirella Lapata. Models of semantic representation with visual attributes. In *Proc. 51st Ann. Meeting of the Association for Computational Linguistics*, ACL ’13, pages 572–582, 2013.
- [59] Ameesh Makadia, Vladimir Pavlovic, and Sanjiv Kumar. A new baseline for image annotation. In *Proc. 10th European Conf. on Computer Vision*, ECCV ’08, pages 316–329. Springer-Verlag, 2008.
- [60] Glen Jeh and Jennifer Widom. Simrank: a measure of structural-context similarity. In *Proc. 8th Int. Conf. on Knowledge Discovery and Data Mining*, KDD ’02, pages 538–543. ACM, 2002.
- [61] Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. Wordnet::similarity: measuring the relatedness of concepts. In *HLT-NAACL: Demonstration Papers*, pages 38–41. Association for Computational Linguistics, 2004.
- [62] Yushi Jing and Shumeet Baluja. Visualrank: applying pagerank to large-scale image search. *Trans. on Pattern Analysis and Machine Intelligence*, 30(11):1877–1890, Nov 2008.
- [63] Jingwen Wang, Jianlong Fu, Yong Xu, and Tao Mei. Beyond object recognition: Visual sentiment analysis with deep coupled adjective and noun neural networks. In *IJCAI*, pages 3484–3490, 2016.
- [64] Takuya Narihira, Damian Borth, Stella X Yu, Karl Ni, and Trevor Darrell. Mapping images to sentiment adjective noun pairs with factorized neural nets. *arXiv preprint arXiv:1511.06838*, 2015.
- [65] Quanzeng You, Jiebo Luo, Hailin Jin, and Jianchao Yang. Robust image sentiment analysis using progressively trained and domain transferred deep networks. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI’15, pages 381–388. AAAI Press, 2015.

- [66] Tao Chen, Damian Borth, Trevor Darrell, and Shih-Fu Chang. Deepsentibank: Visual sentiment concept classification with deep convolutional neural networks. *arXiv preprint arXiv:1410.8586*, 2014.
- [67] Damian Borth, Rongrong Ji, Tao Chen, Thomas Breuel, and Shih-Fu Chang. Large-scale visual sentiment ontology and detectors using adjective noun pairs. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 223–232. ACM, 2013.
- [68] Naila Murray, Luca Marchesotti, and Florent Perronnin. Ava: A large-scale database for aesthetic visual analysis. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2408–2415. IEEE, 2012.
- [69] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, Fabian Nater, and Luc Van Gool. The interestingness of images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1633–1640, 2013.
- [70] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [71] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [72] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. What makes paris look like paris? *ACM Transactions on Graphics*, 31(4), 2012.
- [73] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.
- [74] Zhangyang Wang, Jianchao Yang, Hailin Jin, Eli Shechtman, Aseem Agarwala, Jonathan Brandt, and Thomas S Huang. Deepfont: Identify your font from an image. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 451–459. ACM, 2015.

References

- [75] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.
- [76] Yushi Jing and Shumeet Baluja. Visualrank: Applying pagerank to large-scale image search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11):1877–1890, 2008.
- [77] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *AAAI*, pages 2852–2858, 2017.
- [78] Cesc C Park and Gunhee Kim. Expressing an image stream with a sequence of natural sentences. In *NIPS*, pages 73–81, 2015.
- [79] Xin Wang, Wenhui Chen, Yuan-Fang Wang, and William Yang Wang. No metrics are perfect: Adversarial reward learning for visual storytelling. *arXiv preprint arXiv:1804.09160*, 2018.
- [80] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in Neural Information Processing Systems*, pages 3483–3491, 2015.
- [81] Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. Every picture tells a story: Generating sentences from images. In *ECCV*, pages 15–29, 2010.
- [82] Andrej Karpathy, Armand Joulin, and Fei Fei Li. Deep fragment embeddings for bidirectional image sentence mapping. In *NIPS*, pages 1889–1897, 2014.
- [83] Girish Kulkarni, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C Berg, and Tamara L Berg. Baby talk: Understanding and generating image descriptions. In *CVPR*, pages 1601–1608, 2011.
- [84] Xinlei Chen and C Lawrence Zitnick. Mind’s eye: A recurrent visual representation for image caption generation. In *CVPR*, pages 2422–2431, 2015.
- [85] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. In *CVPR*, pages 4651–4659, 2016.

- [86] Ting Yao, Yingwei Pan, Yehao Li, Zhaofan Qiu, and Tao Mei. Boosting image captioning with attributes. In *IEEE International Conference on Computer Vision, ICCV*, pages 22–29, 2017.
- [87] Qiuyuan Huang, Zhe Gan, Asli Celikyilmaz, Dapeng Wu, Jianfeng Wang, and Xiaodong He. Hierarchically structured reinforcement learning for topically coherent visual story generation. *arXiv preprint arXiv:1805.08191*, 2018.
- [88] Jack Hopkins and Douwe Kiela. Automatically generating rhythmic verse with neural networks. In *ACL*, volume 1, pages 168–178, 2017.
- [89] Rui Yan, Han Jiang, Mirella Lapata, Shou-De Lin, Xueqiang Lv, and Xiaoming Li. i, poet: Automatic chinese poetry composition through a generative summarization framework under constrained optimization. In *IJCAI*, pages 2197–2203, 2013.
- [90] Marjan Ghazvininejad, Xing Shi, Yejin Choi, and Kevin Knight. Generating topical poetry. In *EMNLP*, pages 1183–1191, 2016.
- [91] Xingxing Zhang and Mirella Lapata. Chinese poetry generation with recurrent neural networks. In *EMNLP*, pages 670–680, 2014.
- [92] Linli Xu, Liang Jiang, Chuan Qin, Zhe Wang, and Dongfang Du. How images inspire poems: Generating classical chinese poetry from images with memory networks. In *AAAI*, 2018.
- [93] Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought vectors. In *NIPS*, pages 3294–3302, 2015.
- [94] Hisar Maruli Manurung. A chart generator for rhythm patterned text. In *Proceedings of the First International Workshop on Literature in Cognition and Computer*, pages 15–19, 1999.
- [95] Hugo Oliveira. Automatic generation of poetry: an overview. *Universidade de Coimbra*, 2009.
- [96] Hugo Gonalo Oliveira. Poetryme: a versatile platform for poetry generation. *Computational Creativity, Concept Invention, and General Intelligence*, 1:21, 2012.

- [97] Jing He, Ming Zhou, and Long Jiang. Generating chinese classical poems with statistical machine translation models. In *AAAI*, 2012.
- [98] Long Jiang and Ming Zhou. Generating chinese couplets using a statistical mt approach. In *COLING*, pages 377–384, 2008.
- [99] Marjan Ghazvininejad, Xing Shi, Jay Priyadarshi, and Kevin Knight. Hafez: an interactive poetry generation system. *ACL*, pages 43–48, 2017.
- [100] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.
- [101] Wojciech Zaremba and Ilya Sutskever. Reinforcement learning neural turing machines-revised. *arXiv preprint arXiv:1505.00521*, 2015.
- [102] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *CVPR*, volume 1, page 3, 2017.
- [103] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Tomas Mikolov, et al. Devise: A deep visual-semantic embedding model. In *NIPS*, pages 2121–2129, 2013.
- [104] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*, 2014.
- [105] Jianlong Fu, Heliang Zheng, and Tao Mei. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In *CVPR*, pages 4438–4446, 2017.
- [106] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *NIPS 2014 Workshop on Deep Learning*, 2014.
- [107] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002.
- [108] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015.

References

- [109] Limin Wang, Sheng Guo, Weilin Huang, and Yu Qiao. Places205-vggnet models for scene recognition. *arXiv preprint arXiv:1508.01667*, 2015.
- [110] Andreas Krause and Carlos Guestrin. Near-optimal observation selection using submodular functions. In *Proc. 22nd Nat. Conf. on Artificial Intelligence*, AAAI'07, pages 1650–1654. AAAI Press, 2007.

SELECTED LIST OF PUBLICATIONS

- **Journals**

- [1] Bei Liu, Makoto P. Kato and Katsumi Tanaka. Cognition-Aware Summarization of Photos Representing Events. *IEICE Transactions on Information and Systems*, 99 (12), pp. 3140-3153, 2016.

- **International Conferences**

- [2] Bei Liu, Jianlong Fu, Makoto P. Kato, Masatoshi Yoshikawa. Beyond Narrative Description: Generating Poetry from Images by Multi-Adversarial Training. *ACM Multimedia Conference (MM)*, 2018. (Oral, best paper)
 - [3] Bei Liu, Makoto P. Kato and Katsumi Tanaka. Finding Photo Sets of Events by Minimizing Misrecognition from Neighbor Events. *Proc. of the 15th International Conference on Web-Age Information Management (WAIM)*, 2014.

- **Domestic Conferences and Workshops**

- [4] Bei Liu, Makoto P. Kato and Katsumi Tanaka. Learning Subjective-Adjectives from Images by Stacked Convolutional Auto-Encoders. *The 9th Forum on Data Engineering and Information Management (DEIM)*, G6-1, 2017.
 - [5] Bei Liu, Makoto P. Kato and Katsumi Tanaka. Estimating Desired Actions Stimulated by Annotated Images. *The 8th Forum on Data Engineering and Information Management (DEIM)*, A4-2, 2016.
 - [6] Bei Liu, Makoto P. Kato and Katsumi Tanaka. Estimating Interestingness of Images based on Viewer Data. *The 7th Forum on Data Engineering and Information Management (DEIM)*, A5-3, 2015.

Selected List of Publications

- [7] Bei Liu, Makoto P. Kato and Katsumi Tanaka. Finding Photo Sets of Events by Minimizing Misrecognition from Neighbor Events. *The 6th International Workshop with Mentors on Databases, Web and Information Management for Young Researchers*, 2014.
- [8] Bei Liu, Makoto P. Kato and Katsumi Tanaka. Finding Photo Sets of Events by Differentiation from Neighbors. *WebDB Forum*, 2013.

APPENDIX

A Proof for Section 3.4.6

As stated, Equation 3.6 is NP-hard, which prevents us from finding the best image set that can maximize our function. Fortunately, this function is rich in structure, enables a greedy algorithm to be applied, and solves the problem by finding a good approximation to the optimum. We prove the submodularity and monotonicity of the objective function that admits the greedy algorithm.

A.1 Proof of Submodularity

According to [56], submodularity can be defined as follows.

Definition 14 (Submodularity). *Given a finite ground set N , a set function $2^N \rightarrow \mathbb{R}$ is submodular if and only if for all sets $S, T \subset N$ such that $S \subset T$, and $d \in N \setminus T$, $f(S \cup \{d\}) - f(S) \geq f(T \cup \{d\}) - f(T)$.*

Lemma 2. *$f(S, e)$ is a submodular function.*

Proof. Intuitively, the objective function is submodular because an image set would have already conveyed an event to the user, and therefore, the incremental gain for an additional image is smaller. Let us now prove the submodularity mathematically.

The class of submodular functions is closed under non-negative linear combinations based on the property of submodularity [110]. Three criteria in the function are all non-negative; therefore, to prove the submodularity of function f , we only need to prove the submodularity of each function individually. As proved in [54], functions $\text{SubCov}(S, e)$ and $\text{SupCov}(S, e)$ are submodular. Let S, T be two arbitrary sets of images related by

$S \subset T$. Let m be an image not in T . S' denotes $S \cup \{m\}$, and T' for $T \cup \{m\}$.

$$\begin{aligned} \text{SibDif}(S', e) - \text{SibDif}(S, e) = \\ \prod_{s \in S} \max_{v \in \text{Sib}(e)} P(c = 1 | s, v) \cdot (1 - \max P(c = 1 | m, v)) \end{aligned} \quad (7.1)$$

Similarly, the following formula is workable:

$$\begin{aligned} \text{SibDif}(T', e) - \text{SibDif}(T, e) = \\ \prod_{s \in T} \max_{v \in \text{Sib}(e)} P(c = 1 | s, v) \cdot \prod_{s \in T \setminus S} \max_{v \in \text{Sib}(e)} P(c = 1 | s, v) \cdot (1 - \max P(c = 1 | m, v)) \end{aligned}$$

For each image s and an event v , $P(c = 1 | s, v)$ is the probability that s covers v , which means $P(c = 1 | s, v)$ has a value between 0 and 1. Thus,

$$\prod_{s \in T \setminus S} \max_{v \in \text{Sib}(e)} P(c = 1 | s, v) \leq 1 \quad (7.2)$$

Therefore, we conclude that

$$\text{SibDif}(S', e) - \text{SibDif}(S, e) \geq \text{SibDif}(T', e) - \text{SibDif}(T, e) \quad (7.3)$$

As a result, the function $\text{SibDif}(S, e)$ is submodular, and thus objective function $f(S, e)$ is also submodular. □

A.2 Proof of Monotonicity

In calculus, if a function is monotonically increasing, it should satisfy the following condition:

Definition 15 (Monotonic Increasing). *Given a finite ground set N , a set function $2^N \rightarrow \mathbb{R}$ is monotonic increasing if and only if for all sets $A, B \subset N$, $A \subset B$, $f(A) \leq f(B)$.*

Intuitively, a function is monotonic increasing when the function between ordered sets can preserve the given order. In our case, with more images in an image set, the perceptual quality of the image set is higher.

Lemma 3. *$f(S, e)$ is monotonic increasing.*

Proof. According to the definition of monotonic increasing, a non-negative linear combination of monotonic increasing functions is also monotonic increasing based on the property of adding inequalities (if $a > b$ and $c > d$, then $a + c > b + d$). Let us prove each of the three criteria one by one. Let S, T be two arbitrary sets of images related by $S \subset T$. On the basis of the above demonstration, we can easily get:

$$\begin{aligned} \text{SubCov}(T, e) - \text{SubCov}(S, e) = & \sum_{v \in \text{Sub}(e)} P(v|e) \cdot \left(\prod_{s \in S} (1 - P(c = 1|s, v)) \right) \\ & \cdot \left(1 - \prod_{s \in T \setminus S} (1 - P(c = 1|s, v)) \right) \end{aligned} \quad (7.4)$$

Since the probability of an image that covers an event is between 0 and 1, we can obtain:

$$\text{SubCov}(T, e) - \text{SubCov}(S, e) \geq 0 \quad (7.5)$$

Thus, $\text{SubCov}(S, e)$ and $\text{SupCov}(S, e)$ are confirmed to be monotonic increasing. In the case of difference from sibling-events $\text{SibDif}(s, e)$, we can have:

$$\begin{aligned} \text{SibDif}(T, e) - \text{SibDif}(S, e) = & \prod_{s \in T} \max_{v \in \text{Sib}(e)} P(c = 1|s, v) \cdot \left(1 - \prod_{s \in T \setminus S} \max_{v \in \text{Sib}(e)} P(c = 1|s, v) \right) \end{aligned} \quad (7.6)$$

As clarified in the part where we prove submodularity, $P(c = 1|s, v)$ is between 0 and 1 for every image s and event v , so the above formula can be deduced to be non-negative, i.e.:

$$\text{SibDif}(T, e) - \text{SibDif}(S, e) \geq 0 \quad (7.7)$$

Hence, function $\text{SibDif}(s, e)$ is monotonic increasing as desired. We can safely draw the conclusion that $f(S, e)$ is monotonic increasing.

With the satisfaction of submodularity and monotonicity, we eventually confirm that the greedy algorithm is a proper approach to generate an approximate image set to the optimum image set.

□