Sentiment Analysis on Multi-view Social Data

by

Teng Niu

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the M.C.S. degree in
Computer Science

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

With the proliferation of social networks, people are likely to share their opinions about news, social events and products on the Web. There is an increasing interest in understanding users' attitude or sentiment from the large repository of opinion-rich data on the Web. This can benefit many commercial and political applications. Primarily, the researchers concentrated on the documents such as users' comments on the purchased products. Recent works show that visual appearance also conveys rich human affection that can be predicted. While great efforts have been devoted on the single media, either text or image, little attempts are paid for the joint analysis of multi-view data which is becoming a prevalent form in the social media. For example, paired with the posted textual messages on Twitter, users are likely to upload images and videos which may carry their affective states. One common obstacle is the lack of sufficient manually annotated instances for model learning and performance evaluation. To prompt the researches on this problem, we introduce a multi-view sentiment analysis dataset (MVSA) including a set of manually annotated image-text pairs collected from Twitter. The dataset can be utilized as a valuable benchmark for both single-view and multi-view sentiment analysis. In this thesis, we further conduct a comprehensive study on computational analysis of sentiment from the multi-view data. The state-of-the-art approaches on single view (image or text) or multi-view (image and text) data are introduced, and compared through extensive experiments conducted on our constructed dataset and other public datasets. More importantly, the effectiveness of the correlation between different views is also studied using the widely used fusion strategies and advanced multi-view feature extraction methods.

Index Terms: Sentiment analysis, social media, multi-view data, textual feature, visual feature, joint feature learning.

Acknowledgements

I would like to give my sincerest gratitude and appreciation to my supervisor, Prof. Abdulmotaleb El Saddik, for his continuous guidance and support not only in academic domain but also in my personal life.

Unique and sincere thanks go to Dr. Shiai Zhu for the precious assistance, invaluable guidance, and feedback he supplied going through my research, as well as his review and revision of this thesis. I also need to thank to Mr. Lei Pang for helping me on the implementation of some algorithms.

I would also like to thank all my colleagues in the MCRLab for their suggestions and contributions throughout the research, and all my friends for their help on my campus life. I list some of them here: Haiwei Dong, Lin Yang, Jun Yang, Yangyang Liu, Hao Kuang, Longyu Zhang, Minggao Wei, Bote Han, Yu Gao, Chang Li, Zhongli Li, Bowen Yang and Jing Wang.

Finally, I am grateful to my family. Their love, understanding and support made me grow up these years.

Table of Contents

List of Tables

List of Figures

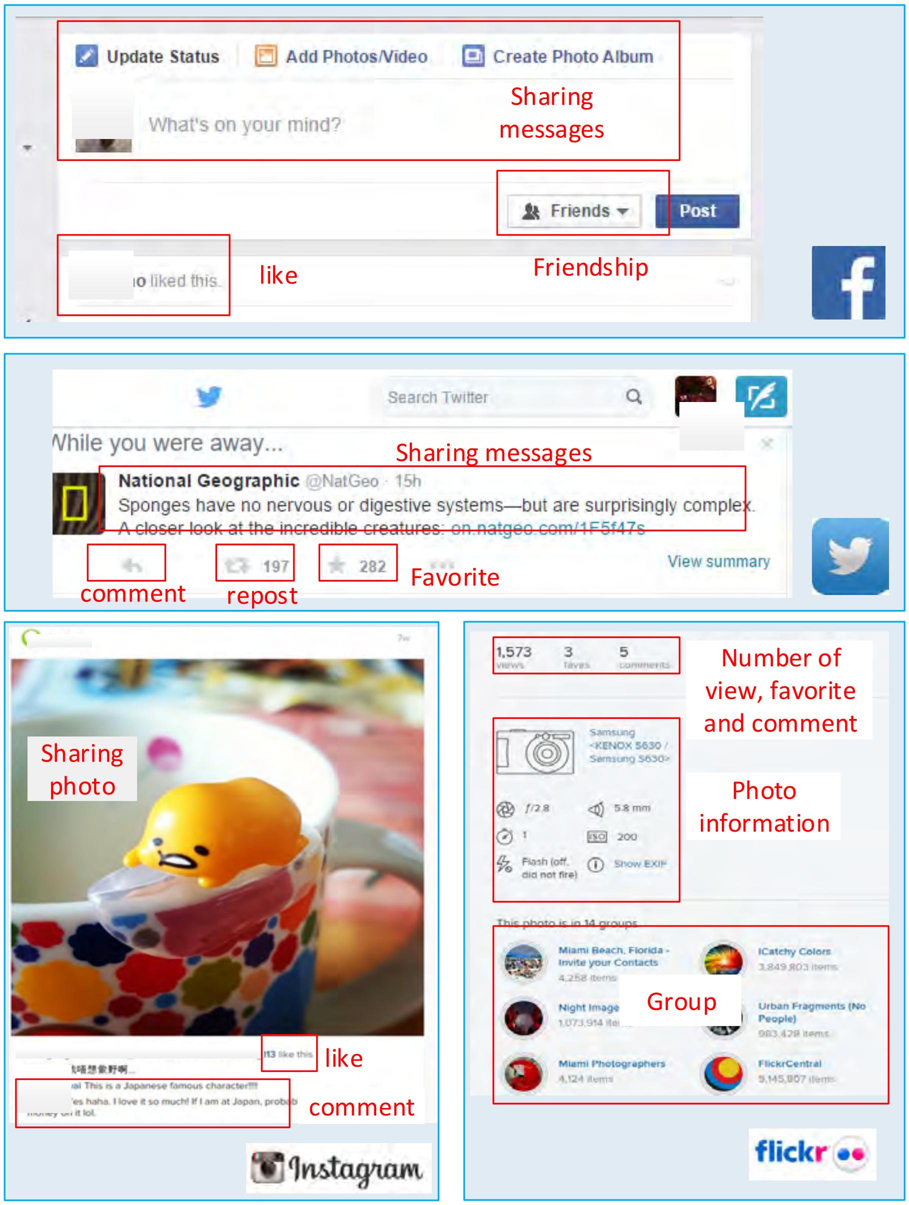
Chapter 1

Introduction

1.1 Background and Motivation

1.1.1 Social Media

Social Website provides a convenient platform, on which users can share information with their friends and post their timely status or attitude. There exits the most up-to-date and heterogeneous data of users. As showed in Figure 1.1, different social websites provide various functions that users can upload, comment and repost messages with different media types. In addition, friendship and community can also be built on the Web. We can obtain plenty of knowledge from the users who contributed data and social activities. Many applications benefit from the exploration of such rich resource, ranging from media data understanding to big data visualization. For example, tagged images and videos can be utilized as weakly (inaccurate) labeled training instances for classifier learning. Recommendation can be performed by exploring the common patterns embedded in the crowd-sourcing knowledge of Web users. In addition, geo-location based services can benefit from the plenty of geo-tagged social data. For each user, personal interests can



Sponges have no nervous or digestive systems—but are surprisingly complex. A closer look at the incredible creatures: on.natgeo.com/1F5f47s

唔想食野啊...

al This is a Japanese famous character!!!!

'es haha. I love it so much! If I am at Japan, probab money on it lol.

Figure 1.1: Popular social networks on the Web: Facebook, Twitter, Instagram and Flickr. Plenty of data is generated with the various user social activities using the rich functions (marked by red boxes) provided by social media.

be identified from the history of user social activities. This is especially important for providing personalized services, such as image tagging, search and recommendation.

On the other hand, social media poses many challenges. Firstly, the data contributed by general users tends to be diverse, inaccurate and unstructured. Mining knowledge from these data needs careful designs. Secondly, with the daily updated huge data, all the applications and designed algorithms should consider the problem of scalability. Efficient solution is extremely important for analyzing social media. Thirdly, the data on the Web has quite different attributes. It is difficult to represent them in a unified way. For example, image can be represented as visual features, while a user is usually represented as some statistical features such as the number of uploaded images or comments. In addition, the social entities are connected with each other, finally a huge network can be extracted. This further introduces significant difficulties on social media analysis. Other interesting issues in social media include the message diffusion on the Web, data organization and visualization, etc.

1.1.2 Sentiment Analysis

One of the most important aspects in social data is the conveyed human affection, which is the focus of this thesis. Companies can investigate the consumers' opinions on their products or design new marketing strategies based on the discussions on the products in social media. For politicians, the attitude of voters exploited from social media is helpful for predicting the result of an election campaign. A fundamental issue behind these prevalent and anxious needs is to accurately analyze sentiment carried in the user generated data, which has recently enjoyed eruption of research activity.

Sentiment analysis aims at computationally identifying people's opinion or attitude towards entities such as events, topics or product features. This work is also known as opinion mining or subjectivity analysis which are used interchangeably in literature. In [43], the differences between sentiment analysis and other researches on affective computing, such as emotion detection and mood analysis, are discussed. In general, the computational representation of sentiment can be either categorical states (i.e., two opposing sentiment polarities) or the level of positivity (i.e., continues value between two polarities). In this thesis, we will concentrate on the problem of filling data into positive or negative sentiment as categorical states are easier for people to understand and justify. This is essential for many applications such as opinion-oriented summarization and item retrieval.

Sentiment analysis can be performed on different kinds of media types, such as text, image or video. Sentiment analysis of textual document has been a longstanding research field. We can roughly categorize existing works into two groups: lexicon-based approaches and statistic learning approaches. The former leverages a set of pre-defined opinion words or phrases, each of which is assigned with a score representing its positive or negative level of sentiment. Sentiment polarity is the aggregation of opinion values of terms within a piece of text. Two widely used sentiment lexicons are SentiStrength[1] and SentiWordnet[2]. On the other hand, statistic learning approaches treat sentiment analysis as a standard classification problem. A variety of supervised learning approaches are utilized with some dedicated textual features. In addition, sophisticated nature language processing (NLP) techniques such as dependency parsing are developed to address the problems of syntax, negation and irony. These techniques can be found in two comprehensive surveys [43, 29]. On the other hand, visual sentiment analysis attracted extensive attentions recently, as visual instances is exponentially increasing in social media. The basic idea on visual sentiment analysis follows the same way of automatic visual content understanding.

[1] http://sentistrength.wlv.ac.uk/

[2] http://sentiwordnet.isti.cnr.it/

1.1.3 Predicting Sentiment in Social Data

Previous efforts concentrate mostly on opinionated textual documents from review-aggregation resources such as Internet-based retailers and forum Websites, where the fetched texts are usually in stereotyped format and substantially different from the unstructured data on social Websites. As a platform for quick and instant information sharing, Twitter messages are short, and thus difficult to gather sufficient statistics for sophisticated sentiment analysis. In addition, the desired material is regarded as "dirty", since users' presentation may be quite different in style, content and vocabulary (e.g., abbreviations or acronym). Thus, it is much harder for machines to accurately analyze these free-form texts [83]. To address this problem, recent works exploit additional information in social media, such as user relationship [19, 61] or Hashtags [28]. However, this is a subtask for boosting the performance rather than a standalone problem. In this thesis, we will study the fundamental issues of sentiment analysis using traditional techniques, whose performances on Twitter messages are still unclear. This is different from previous studies [44, 43], which investigate this problem on well-structured documents.

As a new and active research area, visual sentiment analysis adopts a similar framework with general concept detection. Supervised learning techniques are utilized on extracted low-level (e.g., GIST and SIFT) or middle-level visual features [66, 26]. In [79, 74], robust feature learning using deep neural networks is introduced into sentiment analysis. Similar to the semantic gap in concept detection, there is also an affective gap in sentiment analysis. To narrow down the gap, middle-level features defined on a set of affective atom concepts [81] and emotional Adjective Noun Pairs [3, 5] are investigated.

To this end, sentiment analysis is performed purely on textual or visual data. However, Web users tend to post messages containing different media types. For example, Twitter messages are usually attached with images or videos, which are more attractive than texts [78]. The statistical analysis in [78] shows that there is positive correlation between the sentiment detected from texts and images. Thus more accurate results are expected by taking different views into consideration. The challenge coming up with this advantage is to analyze the data types with quite different characteristics. Early or late fusion, which simply combines the results generated from different views, is the most straightforward way [22, 3]. In [77], emotional images are selected by integrating the rank lists produced using visual content and corresponding texts respectively. However, the interaction between different views is ignored. While little effort has been devoted on sentiment analysis of multi-view data, many researches on cross-view or multi-view learning [73, 23] may be helpful to handle this problem. For example, a joint representation of multi-view data is developed using Deep Boltzmann Machine (DBM) in [60]. In this thesis, the different ways for combining the information from multiple views will be introduced and evaluated.

Since supervised learning needs plenty of clean training instances, one important obstacle causing the limited progress of sentiment analysis in social media is the insufficient manually justified training data, especially for visual and multi-view analysis. In addition, to promote the research on this problem, a challenging benchmark dataset is needed. In [3], a few hundreds of labeled multi-view Twitter messages are provided for sentiment analysis. This small dataset is further used in [74] for transferring Convolutional Neural Networks (CNN) trained on object images into the domain of sentiment classification. In [79], textual analysis results are utilized as weak labels for pre-training CNN, which is further turned using annotated 1,269 Twitter images. Other efforts on dataset construction for affective computing include video emotion detection [22] and aesthetics analysis [63, 38]. These datasets are either too small or not able to be directly used for sentiment analysis.

1.2 Objective and Contribution

We target at identifying user opinion from the posted messages with different types of media. To release the constraint on benchmark datasets, we contribute a set of multi-view

Twitter messages annotated with sentiment polarity information. Furthermore, state-of-the-art approaches are investigated. In summary, we make three important contributions in this thesis:

- To prompt the research on the multi-view sentiment analysis, we develop a benchmark dataset including manually justified twitter messages. The labels are provided for both text and image in each message. The provided dataset is larger than other multi-view dataset for sentiment analysis. The researcher can generate their own datasets based our provided data and annotations. In addition, the annotation tools are public accessible. Thus the dataset can be updated regularly with the upcoming annotations contributed by the users.

- We analyzed on the state-of-the-art approaches about textual, visual and multi-view sentiment analysis. Some insightful discussions on each of the approaches are given. Furthermore, we provide a comprehensive study of sentiment analysis on textual and visual data respectively. The feasibility of different methods on Twitter messages is evaluated through a comprehensive set of experiments conducted on public available datasets and our constructed dataset. Several aspects which may affect the performance are experimented and compared.

- Finally, we propose to predict sentiment from multi-view social data by exploring the correlation between textual and visual information carried in the Twitter messages, which is rarely considered in previous works. Both the traditional fusion strategies and the most advanced multi-view feature learning approaches are illustrated and experimentally studied. Eventually, we provide a pipeline for detecting sentiment in social data.

1.3 Scholarly Achievements

The major parts of the work in this thesis have been accepted by the following publication:

- **Teng Niu**, Shiai Zhu, Lei Pang and Abdulmotaleb El Saddik, Sentiment Analysis on Multi-view Social Data, International Conference on Multimedia Modelling (MMM), Miami, USA, January, 2016.

1.4 Thesis Overview

The remain of this thesis is organized as follows:

- Chapter 2 reviews the related works on sentiment analysis including the published datasets, various sentiment analysis approaches on textual, visual and multi-view data.

- Chapter 3 introduces the pipeline for sentiment analysis. Each component in the framework is illustrated in detail. We will focus on the features generally used or dedicatedly designed for our task. Finally, the statistical learning methods wildly used in sentiment analysis are also introduced.

- Chapter 4 describers the process for constructing our dataset. The way for collecting Twitter messages and the designed annotation tool are explained. The statistical analysis on the annotation results of our collected dataset is given in this chapter.

- Chapter 5 evaluates the introduced pipeline using some representative approaches discussed in Chapter 3. Sentiment analysis is performed on single and multi-view data respectively. Several influential factors in the pipeline are experimented and insightful discussions are given.

- Chapter 6 concludes this thesis. The potential future works on data collection and some interesting remaining issues for further study are included.

Chapter 2

Related Work

In this chapter, we first introduce related datasets for sentiment analysis, and then briefly discuss some representative approaches on single-view and multi-view data.

2.1 Sentiment Analysis Datasets

Coming up with the extensive research on text sentiment analysis, there is plenty of datasets available on the Web. These datasets are usually constructed for specific domains. In [76], a dataset is constructed by crawling 2,820,059 news pages from the Web. Another opinion dataset for news sources is MPQA[1], where 535 news posts are labeled at the sentence level and sub-sentence level for various private conditions such as faiths and sentiments [71, 72]. Besides the news, user reviews and comments on the Web are also explored in some datasets. In [33], 50,000 movie reviews with annotations for positive and negative sentiment are provided. Pang and Lee [41, 42] proposed several movie-review datasets including a sentiment polarity dataset, a sentiment-scale dataset and a subjectivity dataset. Both document-level and sentence-level labels are provided. In addition,

[1] http://www.cs.pitt.edu/mpqa/databaserelease/

rating scores provided by Web users are used as sentiment scale. They further introduced another dataset using Yahoo search results corresponding to 69 queries containing keyword "review". These queries are derived from real MSN users' queries created by the 2005 KDD Cup competition [27]. In [17], a set of customer reviews on 5 electronics items downloaded from Cnet and Amazon is manually annotated at sentence level for sentiment analysis. In [10], a more complete economy database[2] is provided. There are three parts in this dataset. The first one consists of feedbacks from Amazon merchants. The second set includes quotation and transaction premiums. The third one is the emotion scores on frequently appeared evaluation phrases on Amazon. In [58], a fine-grained dataset including multiple-aspect restaurant reviews is constructed. It includes 4,488 reviews, which are assigned a 1-to-5 rating for five diverse fields: ambiance, food, service, value and overall experience. A partially annotated dataset for multi-domain sentiment analysis is proposed in [2], which includes stuff reviews from wide variety of product categories on Amazon. Other datasets designed for different applications include congressional floor-debate transcripts [65], NTCIR multilingual dataset [52] extracted from Chinese, Japanese and English blogs, and Blog6 dataset covering many themes.

Recently, as many researchers turned their attentions to the more timely and convenient social data, some corresponding datasets are proposed. A widely used one is STS [11], where training set consists of 1.6 million tweets automatically annotated as positive and negative based on emotion icons (e.g., ":)" or "=)"), and testing set includes 498 manually labeled tweets. While STS is relatively large, the labels are derived from unreliable emotion icons. In [48], a large dataset including manually labeled 20K tweets is constructed for the annually organized competition in SemEval challenge. In these datasets, each message is attached with one label. However, each tweet may include mixed sentiments. In STS-Gold [49], both message-level and entity-level sentiments are assigned to 2,206 tweets and 58 entities. Each tweet is manually annotated with one of five categories, Negative, Positive,

[2] http://economining.stern.nyu.edu/datasets.html

Neutral, Mixed and Other.

Besides the datasets for general sentiment analysis, there are other datasets constructed for specific domains or topics using Twitter messages. Health Care Reform (HCR) dataset [59] is constructed for eight subjects, such as health care reform, Obama, Democrats, Republicans. Sanders dataset[3] includes 5,513 tweets for four topics: Apple, Google, Microsoft and Twitter. In [53], Obama-McCain Debate (OMD) database was created from more than 3,000 tweets posted during the first America presidential TV argument in 2008. Each message in this dataset was annotated by several annotators using Amazon Mechanical Turk (AMT). As reported in [8], the inter-annotator agreement is 0.655, which indicates the effective agreement between annotators. To evaluate the sentiment lexicon SentiStrength, Sentiment Strength Twitter Dataset (SS-Twitter) including 4,242 tweets is introduced in [64]. Each tweet is annotated with a positive score between 1 and 5 or a negative score between -1 and -5. WA, WB and GASP are three sets of tweets drawn from the Dialogue Earth Twitter Corpus [4]. WA (4,490 tweets) and WB (8,850 tweets) are about the weather, and GASP includes 12,770 tweets discussing about gas prices.

Comparing to textual data, very few datasets have been built for sentiment analysis on visual instances. In [79], a total of 1,269 Twitter images (ImgTweet) are constructed for testing their method. Each image is annotated by five annotators from Amazon Mechanic Turk. The ground-truth is generated by considering the number of agreements between the five annotators. It is much more challenging to annotate the multi-view data. In [3], accompanying with their proposed emotional middle level features (SentiANP), a small dataset including 603 multi-view Twitter messages is provided with manual annotations. Originally, 2,000 tweets with both texts and images are showed to annotators also from Amazon Mechanical Turk (AMT). Each message is labeled in three ways, text only, image only and both text and image. Only the message with consistent labels are finally utilized

[3] http://www.sananalytics.com/lab

[4] www.dialogueearth.org

in the benchmark. Besides the texts and images, another related large-scale dataset is proposed in [22], which nevertheless is constructed for emotion detection on Web videos. To the best of our knowledge, there are no other datasets dedicatedly designed for multi-view sentiment analysis.

2.2 Sentiment Analysis Approaches

2.2.1 Text Sentiment Analysis

The existing works on text sentiment analysis can be roughly categorized into two groups: lexicon-based approaches and statistic learning approaches. The former utilizes prior knowledge of opinion words or phrases, which can be either derived from manually constructed knowledge sources or induced from a document corpus. Two widely used sentiment lexicons are SentiStrength and SentiWordnet, where each word or phrase is assigned with a value to indicate its positive or negative strength. Sentiment polarity of a given text can be determined by aggregating the values of opinion phases within it. However, the size of lexicon is relatively small due to the expensive annotation process. In [18], starting with a small set of justified opinion words, the lexicon is enriched by their synonyms and antonyms extracted from WordNet[5]. However, utilizing common knowledge sources (e.g., WordNet) is not able to find domain specific opinion words. Early work in [13] adopts a linguistic approach to extract opinion words which are linked with the available opinion words by conjunctions such as “but” or “and”. In [67], the opinion words are identified if they are frequently co-occurred with “poor” or “excellent” in the documents. In [47], word polarity is determined according to the frequency of the word in positive or negative documents.

On the other hand, sentiment polarity classification is naturally a standard binary

[5] http://wordnet.princeton.edu/

classification problem. A variety of supervised learning approaches can be leveraged with some unique designs. One of the most important aspects is to derive dedicated textual features. A widely used representation is the Bag-of-Words (BoW) model, where each entry in the feature vector corresponds to an individual term (e.g. word or phrase). Despite the various feature selection methods used to find informative terms in traditional text processing, such as Point-wise Mutual Information (PMI) and Chi-square (χ^2), specific approaches for sentiment analysis adopt adjectives and opinion words through part-of-speech tagging [70, 37]. In addition to unigrams which assume tokens are independent, we can define high-order n-grams, where terms are combination of tokens considering their positions in the textual units. Another important aspect is the choice of supervised learning techniques. The most effective approaches investigated in previous works [37, 44, 7] are Naive Bayes (NB), Support Vector Machines (SVM) and Maximum Entropy (ME). However, there is no conclusion on which one is the best, and the choice of feature and learning technique is variant across different domains. For example, unigrams perform better on movie reviews [44], while conversely, Dave et al. [7] reported a better result using bigrams and trigrams on product reviews.

Natural Language Processing (NLP) techniques are usually used as a pre-processing step in lexicon-based approach [36] or linguistic feature extraction. In [4], a deep NLP analysis of the sentences with a dependency parsing stage is used in their proposed sentiment analysis method. In [35], the NLP techniques are used to analyze time and tense presentations. The defined two parameters related to the time of ordering products and the time of using the products. Then, the important opinion threads for the two parameters are extracted from the review data.

2.2.2 Visual Sentiment Analysis

Besides the text, image and video have been two pervasive media types on the Web. The popularity of portable devises integrated with camera and social networks (e.g., Instagram or Twitter) has made the acquisition and dissemination of image/video much more convenient. Computation analysis of human affection expressed in the visual instances is becoming an active research area. A standard way to address this problem leverages the supervised learning techniques with visual features extracted from a set of training data. Inspired by the success of semantic concept detection, which aims at identifying the physical appearance (e.g., object and scene) in visual instances, some handcraft low-level features (e.g., GIST and SIFT) are utilized for visual sentiment analysis in the literature [78, 56, 77]. A well-known problem in concept detection is semantic gap. Similarly, there exists an affective gap between the low-level features and the affection expressed in an image [34]. Meanwhile, different from the concrete concept detection, affective analysis targets at more abstract human concepts. The additional challenges motivate the design of new visual features based on aesthetics, psychological and art theory [34, 24, 82, 63, 69]. In [79, 74], more advanced features learned from large amount of images using deep learning techniques are introduced into sentiment analysis. Unlike the low-level visual features, middle-level features such as Classemes [66] and ObjectBank [26] utilize responses of classifiers trained for a set of atom concepts. These features convey the high-level semantics of visual instances, and are expected to narrow down the semantic gap. Similarly, to bridge the affective gap, affective atom concepts such as sentiment-related attributes [81] and sentiment-aware Adjective Noun Pairs [3, 5] are defined for extracting middle-level features. In this thesis, we will investigate the performances of various features extracted at different levels on the diverse social images.

2.2.3 Multi-view Sentiment Analysis

While great progress has been made on sentiment analysis performed on textual or visual data, little effort is paid on the multi-view social data. A straightforward way [22, 3] is to fuse features or prediction results generated from different views. However, it fails to represent the correlations shared by the multiple views, and thus losses important information for sentiment analysis. The related works on learning cross-view or multi-view representations [73, 23] may be helpful to handle this problem. For example, a joint representation of multi-view data is developed using Deep Boltzmann Machine (DBM) in [60]. However, it is still unclear whether these techniques are able to represent the complex sentiment related context in the multi-view data.

In short, sentiment analysis is intrinsically a binary classification problem. The main difference of different systems lies in the choice of effective feature extraction methods. In addition, a suitable dataset is significantly essential for learning and testing the systems, which may be the reason limiting the needed progress on the sentiment analysis of multi-view data.

2.3 Statistical Learning Algorithms

As we mentioned before, this thesis will focus on predicting sentiment polarity, which is naturally a classification problem. We will only briefly introduce the learning algorithms suitable for our problem. There are many popular statistical learning algorithms that theoretically can be used to solve our problem. Through the extensive studies and researches conducted in different research communities during the last few decades, the most robust and popular methods include Naive Bayes, Maximum Entropy, SVM and so on. Different types of data or features may adopt different models. In addition, the applied domain of data is another factor for selecting the algorithm.

For the text sentiment analysis, Naive Bayes, Maximum Entropy, logistic regression and linear SVM are widely used. Naive Bayes is the simplest probabilistic classifier, which models the posterior probability of a class. Based on Bayes' theorem with naive assumption of independence between words, the problem is converted to model the distribution of each word (feature) in the given class. Maximum entropy is another probability model which is sometimes better than naive Bayes. In maximum entropy, the posterior distribution is directly modeled in an exponential form, where the weighted function is adopted on the extracted features from the instance under context of the target category. Eventually, the optimal parameter is obtained that the entropy of induced distribution is maximized. Note that maximum entropy classification makes no assumption of independence between features. Besides the probability models, both logistic regression and linear SVM are linear models, where the parameters are the weights assigned to feature vectors. The major difference is the adopted loss function. Logistic regression utilizes log loss, while SVM leverages Hinge loss under the principle of maximum marginal classifier. As indicated in [9], these two linear models perform similar in practice. According to the results reported in literature, there is no winner between the discussed algorithms when applied in sentiment or document analysis. In other words, an effective feature representation is essential in text sentiment analysis.

For the visual and multi-view data, linear SVM and logistic regression are usually the optimal choices simply because they can achieve reliable detection accuracy and have the advantage of efficiency. Comparing these two linear model, they have similar properties in effectiveness and efficiency. One advantage of logistic regression is that its prediction results are naturally ranged within [0,1], so that it can be used as the probability. In this way, fusion of multiple models is feasible. For linear SVM, we can convert the decision score into probability using the sigmoid function. Despite the linear model, kernel SVM which has been demonstrated helpful in general recognition tasks is also employed in affective computing. In [22], it is utilized to fuse multiple features by defining several kernels.

However, the disadvantage is that it is computational expensive and is not scalable. Recently, with the development of deep learning architecture, sentiment analysis on image is promoted using deep neural network such as Convolutional Neural Networks [79]. In this thesis, we adopt the linear SVM which has performed promisingly well on text, image and multiple-view data. The detail of this model will be given in section 3.5. Our empirical study on sentiment analysis will focus on the feature aspect which is the most influential component.

Chapter 3

Predicting Sentiment in Multi-view data

This chapter first gives the pipeline of sentiment analysis in section 3.1, which can be utilized for handling different kinds of media data. Then the following sections introduce and compare the detailed techniques designed for different media types, including textual features in section 3.2, visual features in section 3.3, multi-view learning in section 3.4 and the statistical learning strategy in section 3.5.

3.1 Overall Framework

The ultimate goal of sentiment analysis is to understand the polarity or the level of human sentiment from their generated contents. It is intuitively a standard classification or regression problem. Thus, the state-of-the-art sentiment analysis systems follow the basic pipeline showed in Figure 3.1, which is similar to that of many other recognition problems. To learn a classifier for detecting sentiment, we first need a set of labeled training data including both positive and negative instances. Then the instances are represented as

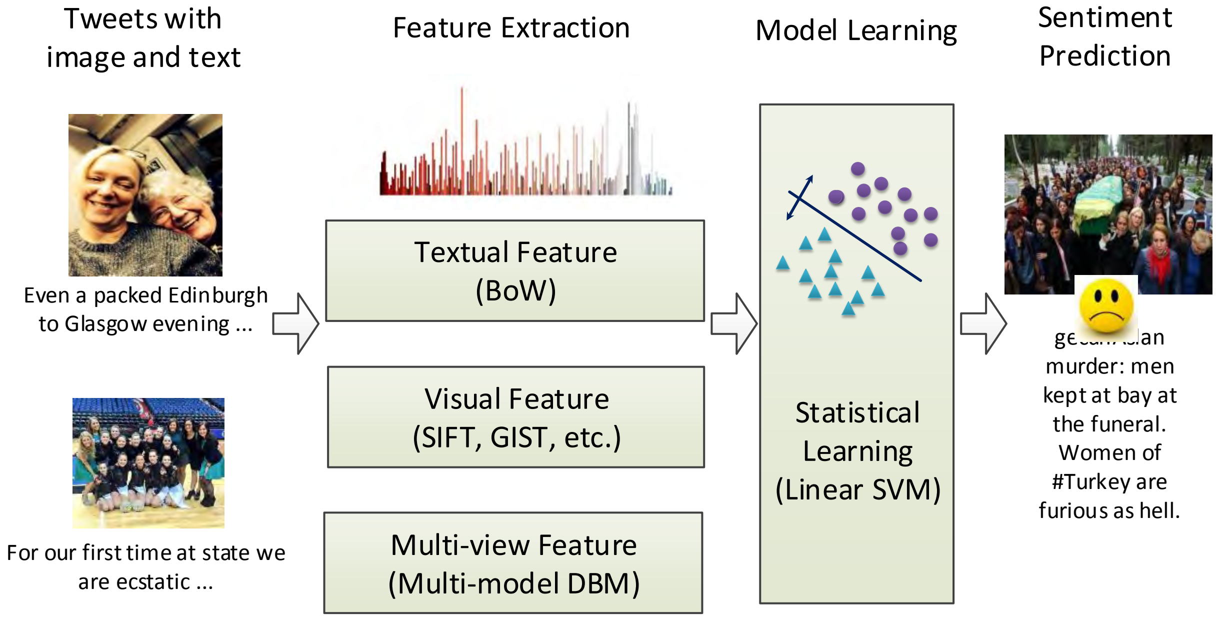

Tweets with image and text
Feature Extraction
Model Learning
Sentiment Prediction
Even a packed Edinburgh to Glasgow evening ...
For our first time at state we are ecstatic ...
Textual Feature (BoW)
Visual Feature (SIFT, GIST, etc.)
Multi-view Feature (Multi-model DBM)
Statistical Learning (Linear SVM)
gecanAslan murder: men kept at bay at the funeral. Women of #Turkey are furious as hell.

Figure 3.1: The pipeline of sentiment analysis on social data, which follows the similar framework in the standard recognition problem.

feature vectors, which are utilized for learning a statistical model (classifier). The model learning can be done off-line. When given a new coming testing instance, the feature is also extracted and the learned corresponding classifier or the regression model is employed for predicting its sentiment.

The two most important components in the pipeline are feature extraction which converts different type of data into feature vectors, and statistical model which learns a classifier or a regression model. In this thesis, we will only consider the classification model as the labels are discrete categories. As for the regression, the techniques are quite similar except the labels may be the scores indicating the level of sentiment. We will introduce each of the components in the following sections. Note that we can also extract different kinds of features representing different aspects of social data. Thus multiple models can be learned for a specific task. The combination of these models can be an additional step in the framework. The discussions on the fusion of multiple features will be also included in this chapter.

3.2 Text-based Approaches

Besides the statistical learning approaches, sentiments in textual documents can also be predicted using lexicons. In this section, we first introduce lexicon-based methods, followed by the textual features used in the statistical model.

3.2.1 Lexicon-based approaches

Lexicon consists of emotional words which are manually assigned with sentiment scores. In this thesis, we consider two popular lexicons: SentiWordnet and SentiStrength as they are the most comprehensive ones.

SentiWordnet is constructed based on the structure of lexical database WordNet[1], where words with the same meaning are grouped into a synset. In SentiWordnet, each synset is associated with three numerical scores corresponding to its level of positive, negative and objective sentiment. As one word may belong to multiple synsets, SentiWordnet further defines which one is the first meaning, second meaning and so on. Given a word, the total sentiment score is computed by weighting its synset scores based on the rank of meanings. SentiWordnet also needs the part-of-speech (PoS) of the given word. In this thesis, we adopt ArkNLP[2], which is designed for parsing Twitter messages. It provides tokenizer, part-of-speech tagger, hierarchical word clusters, and dependency parser for tweets. As showed in Figure 3.2, adjective word "cute" receives positive score 0.54 in SentiWordnet.

[1] https://wordnet.princeton.edu/

[2] http://www.ark.cs.cmu.edu/TweetNLP/

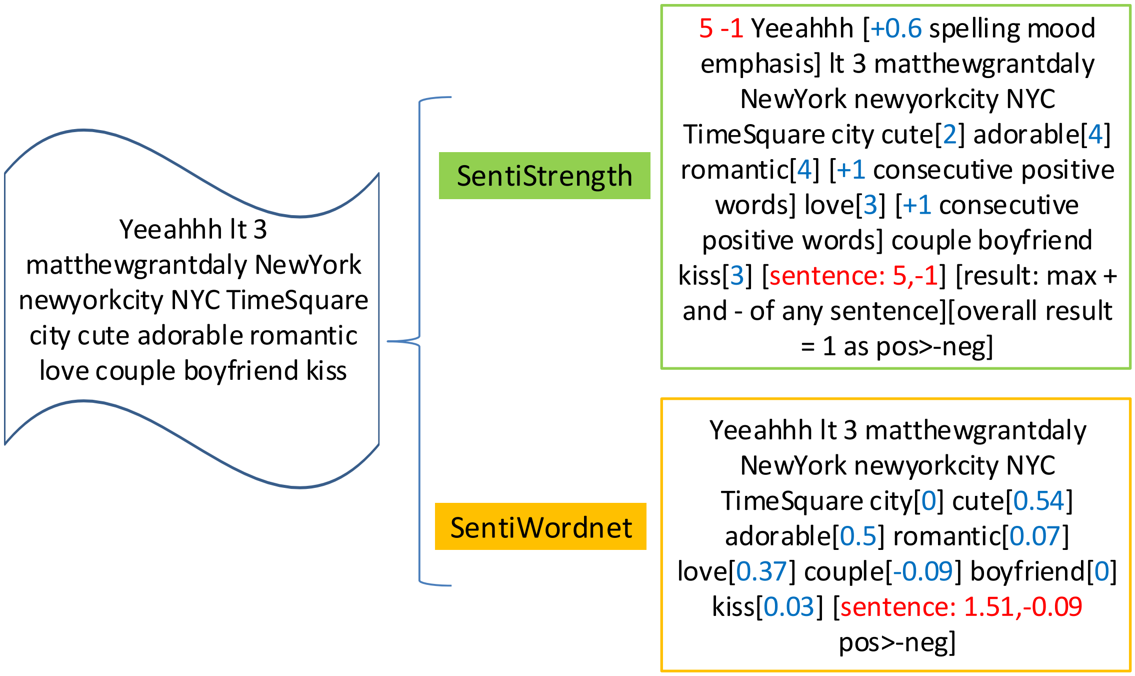

Yeeahhh lt 3 matthewgrantdaly NewYork newyorkcity NYC TimeSquare city cute adorable romantic love couple boyfriend kiss
SentiStrength
5 -1 Yeeahhh [+0.6 spelling mood emphasis] lt 3 matthewgrantdaly NewYork newyorkcity NYC TimeSquare city cute[2] adorable[4] romantic[4] [+1 consecutive positive words] love[3] [+1 consecutive positive words] couple boyfriend kiss[3] [sentence: 5,-1] [result: max + and - of any sentence][overall result = 1 as pos>-neg]
SentiWordnet
Yeeahhh lt 3 matthewgrantdaly NewYork newyorkcity NYC TimeSquare city[0] cute[0.54] adorable[0.5] romantic[0.07] love[0.37] couple[-0.09] boyfriend[0] kiss[0.03] [sentence: 1.51,-0.09 pos>-neg]

Figure 3.2: Sentiment analysis by using two Lexicons: SentiWordnet and SentiStrength. The contents within "[ ]" is the analysis results. Blue parts mean the scores for each individual word, and red parts are the overall results of the sentence.

In SentiStrength, the positive strength of one word is a number between 1 and 5. The meaning of 1 is not positive and 5 means extremely positive. The negative strength score is between -1 and -5. -1 means "not negative" and -5 means "overly negative". In the example of Figure 3.2, positive words include cute, adorable and romantic etc. In addition, the scores of consecutive positive words will be increased by one. Thus the maximum score is 5.

Given a document including N individual words, the overall sentiment score using lexicons can be defined by computing the average scores over all words as:

$$Score = \frac{1}{N}\sum_{n=1}^{N} F_n \times score_n, \tag{3.1}$$

where F_n and $score_n$ are the term frequency and sentiment score of word n respectively. Another way adopted by SentiStrength is defined as:

$$Score = \max_{n \in pos} score_n + \min_{n \in neg} score_n, \tag{3.2}$$

where pos and neg are the word sets received positive and negative scores respectively. Then the sentiment polarity of a document can be decided in the following way:

$$Sentiment = \begin{cases} Positive, & Score > 0 \\ Negative, & Score < 0 \\ Neutral, & Score = 0 \end{cases} \tag{3.3}$$

With the lexicon-based approaches, the message showed in Figure 3.2 is labeled as positive sentiment using either SentiWordnet or SentiStrength.

3.2.2 Textual Features

Bag-of-Words (BoW) is a standard textual feature. The way for generating this feature is showed in Figure 3.3. We first define a vocabulary including a set of terms (individual words or word n-grams) $\mathcal{V} = \{w_1, w_2, \ldots, w_n\}$. The vocabulary is usually constructed by selecting the most frequently appeared terms in the corpus. In addition, other methods can also be utilized for selecting some informative words. For sentiment analysis, the important words can be emotionally related terms, which may be manually selected or simply adjectives identified from the documents using Part-of-speech (PoS) tagging. For general document classification, the informative words can be derived from a large collection of documents corresponding to the target category using statistical methods such as point-wise mutual information (PMI) or Chi-square (χ^2). PMI is defined based on the co-occurrence between target class c and word w. Let $p(w, c)$ be the joint probability of w and c. If the class c and word w have the probabilities $p(c)$ and $p(w)$, PMI is formulated

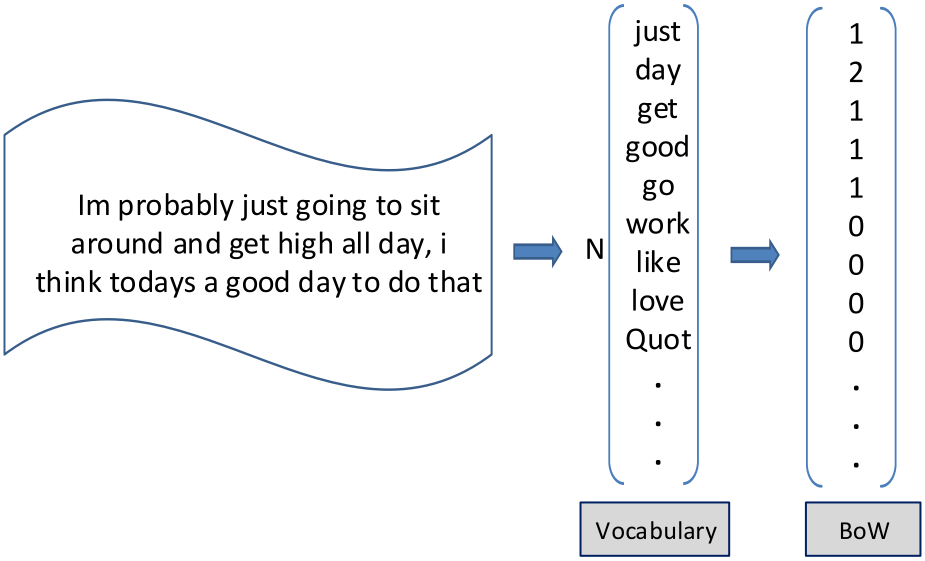

Im probably just going to sit around and get high all day, i think todays a good day to do that
N
just
day
get
good
go
work
like
love
Quot
1
2
1
1
1
0
0
0
0
Vocabulary
BoW

Figure 3.3: The process for extracting Bag-of-Word from a textual message.

as:

$$PMI(w,c) = \log \frac{p(w,c)}{p(c) \times p(w)} = \log \frac{p(w) \times p(c|w)}{p(c) \times p(w)} = \log \frac{p(c|w)}{p(c)}, \tag{3.4}$$

where $p(c|w)$ is the probability of c conditioned on word w. The probability can be easily computed by counting the frequency of each word. $p(w)$ is the fraction of documents containing word w in the whole collection. $p(c|w)$ is the fraction of documents belong to c in the subset of collection containing word w. $p(c)$ is the global fraction of documents from category c. With this definition, word w is positively correlated with c if $PMI(w,c) > 0$. $PMI(w,c) < 0$ indicates the negative correlation between w and c. Both of the positively and negatively correlated words are indicative for recognizing category c.

The correlation between word w and class c using Chi-square (χ^2) is determined by

$$\chi^2(w,c) = \frac{N \times p(w)^2 \times (p(c|w) - p(c))^2}{p(w) \times (1 - p(w)) \times p(c) \times (1 - p(c))}, \tag{3.5}$$

where N is the total number of documents in the collection. According to the definition of Equation 3.5, small $\chi^2(w,c)$ means less association between w and c. In the extreme case that all the documents include w, the χ^2 value becomes zero. We can rank the words based on their χ^2 values. The top ranked words can be utilized for constructing the vocabulary.

With the pre-defined vocabulary, a document is represented as a feature vector, where each element corresponds to one of the words as showed in Figure 3.3. The value of each element can either be a binary value indicating the appearance of the corresponding term or a counting value indicating its term frequency (TF) in the document. In addition, considering the terms may have different importance, each term can be assigned with a weight. One well-known weighting method is inverse document frequency (IDF), which will assign higher weights to words rarely appeared in the documents and penalize frequent words. Formally, it is defined as

$$w^{idf} = log\left(\frac{N}{|\{d : w \in d\}|}\right), \tag{3.6}$$

where N is the total number of documents, and $|\{d : w \in d\}|$ is the number of documents including w. There may be other variants dedicatedly designed for some specific applications. In this thesis, we will evaluate the impacts of some factors in the BoW feature extraction on the text sentiment analysis through the experiments conducted in chapter 5.

Statistics of Text (SoT) has also been proved to be helpful in text sentiment analysis. Basically, some symbols in posted messages may indicate the expressions of users. For example, a question mark may indicate a rhetorical question, and an exclamation mark is an indication that the expressed feeling is emphasized. In [46], nine kinds of text statistics are defined, such as the sum or average of positive and negative scores over all words. The score of each word is derived from the sentiment lexicons (e.g., SentiWordnet). In this thesis, we consider to use the following SoT features:

- The average and sum of positive and negative sentiment scores over all the words. We adopt SentiWordnet and SentiStrength to determine the score of each word.
- The number of question marks "?" in the message.

- The number of exclamation marks “!” in the message.
- The number of combinations of exclamation and question marks “!?” in the message.
- The number of uppercases.

Finally, we can generate a 12 dimensional feature vector by concatenating above statistical values. By using SoT features, the problem of sparsity in BoW generated from short tweets can be addressed to certain extent. It can be used as a complementary to the BoW feature. Comparing to BoW, where vocabulary is derived from a specific domain and may not be suitable for different data collections, one advantage of SoT feature is that the definition is independent to the applied domains.

3.3 Visual-based Approaches

There are a plenty of visual features proposed for different visual recognition tasks. Here we group them into three categories: low-level, middle-level and aesthetic features. As showed in Figure 3.4, low-level features represent the visual content directly from the pixel values by mining the common patterns, informative regions or distribution of colors. The image is treated as a two dimensional signal. One major problem of low-level feature is the so called semantic gap defined as the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation [57]. In contrast, the middle-level feature adopts a set of concept classifiers built upon the low-level features, so that the image can be represented at the semantical level. This kind of features is expected to be helpful for narrowing down the semantic gap. The third category is aesthetic features which are dedicatedly designed for extracting the abstract human perception on the visual appearance. The target is to reduce the so called “affective gap” defined as the lack of coincidence between the measurable signal properties, commonly referred to as features, and the expected affective state in which the

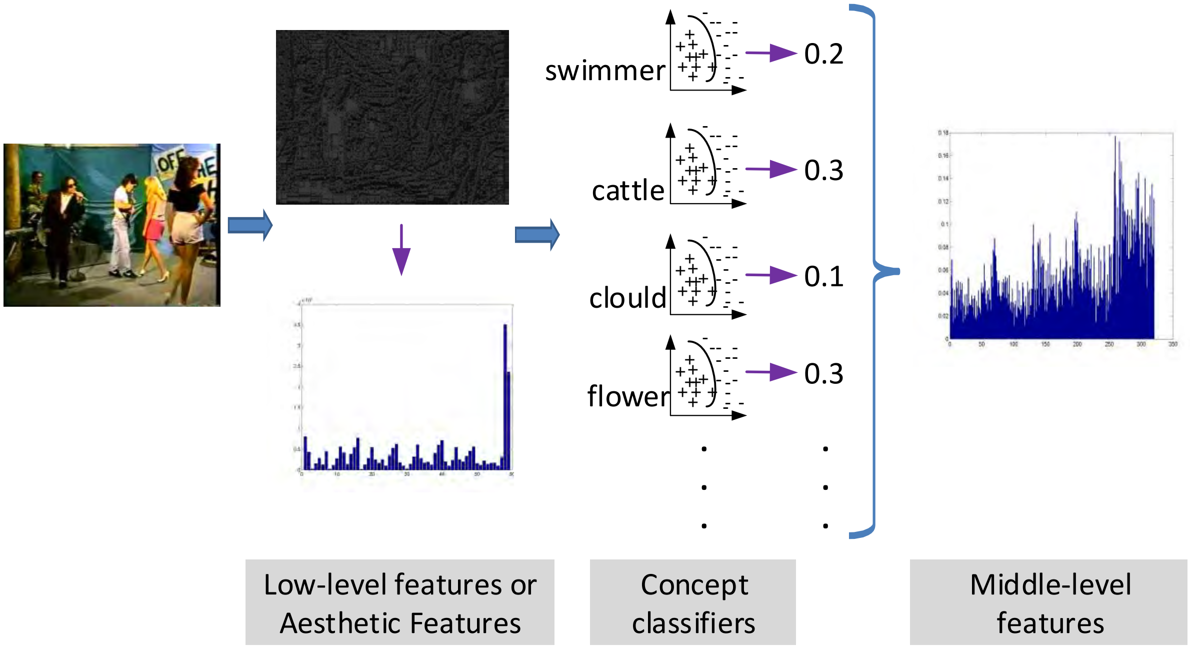


Figure 3.4: Image representations at different levels. Low-level features or aesthetic features are derived from the raw images. Middle-level features consist of the responses of a set of atom concept detectors built upon the low-level features extracted from the images.

user is brought by perceiving the signal [12]. Aesthetic features are also derived from the image signal, but with consideration of human affections when viewing the appearance of an image.

3.3.1 Low-level Visual Features

The low-level features adopted in our work include Color Histogram, GIST, Local Binary Pattern (LBP) and Bag-of-Visual-Words (BoVW).

Color Histogram [54] is the basic visual feature extracted from RGB color channels. For each channel, we can get a 256 dimensional histogram indicating the distribution of

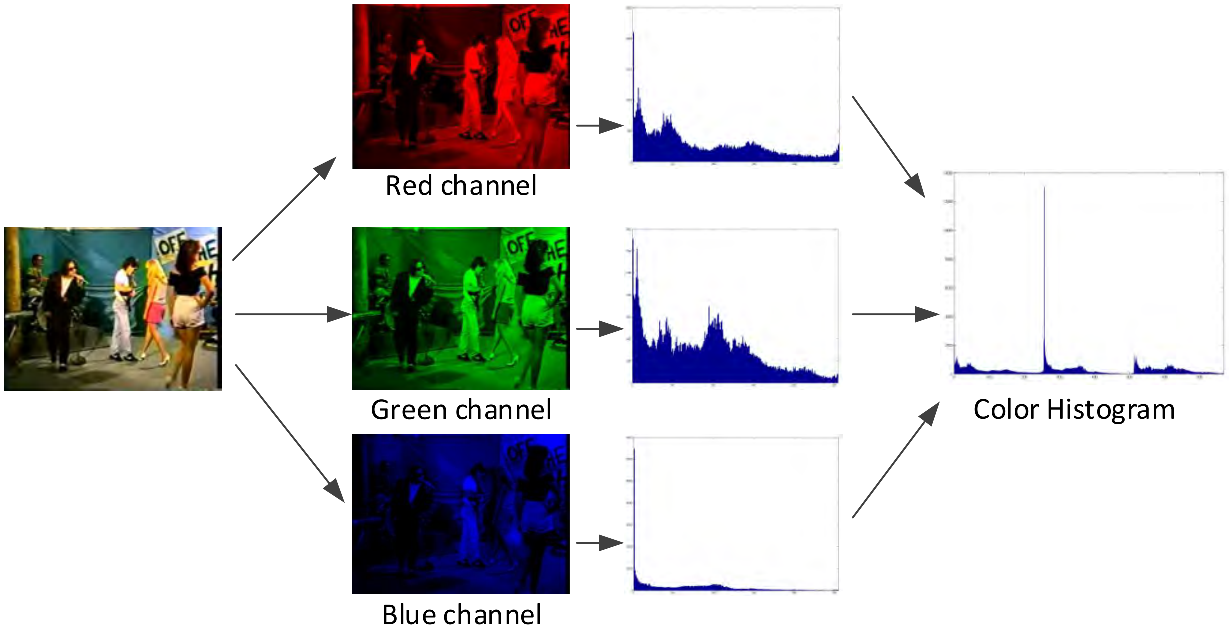


Figure 3.5: The process for extracting color histogram from a single image.

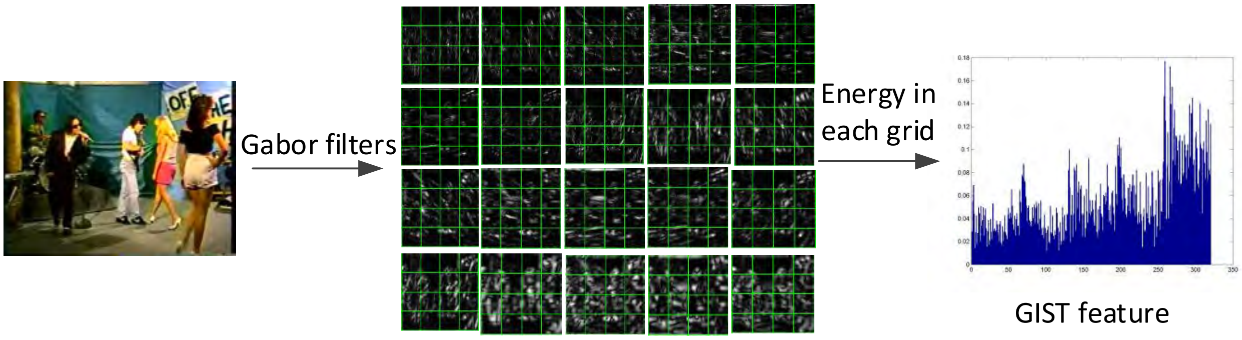


Figure 3.6: The process for extracting GIST feature from a single image.

pixel values. Then the three 256 dimensional histograms are concatenated as a feature vector. After that, the feature vector is normalized to unit length. Figure 3.5 shows the way for generating color histogram.

GIST descriptor [40] is proposed for representing real-world scenes in a computational model. The image is first filtered by using several Gabor filters with different scales and orientations. The texture information in the image can be captured by using the filters. In this thesis, we consider to use 3 scales, the number of orientations under each scale is set as 8, 8 and 4 respectively. In total, we generate 20 feature maps corresponding to the 20 Gabor filters. The maps are showed in Figure 3.6. For each of the map, we split it into 4×4 grids. The average response value of each grid is defined as its energy. Eventually, all the energies are combined into a 320 dimensional GIST feature vector.

Local Binary Pattern (LBP) [14] descriptor is a popular texture feature. Each pixel is represented as binary codes (pattern) by comparing its value with the values of the neighbors. For the example in Figure 3.7, the value of the pixel marked as red is compared with its 8 neighbors. If the value of center pixel is greater than the value of a neighbor, the neighbor is coded as 1, otherwise 0. In this case, we generate a 8 bit code for each pixel. Starting from the left-middle and following the clockwise along a circle, the LBP code for red pixel in Figure 3.7 is “00000110”. In this way, all the pixels are represented to one of the $2^8 = 256$ candidate LBP codes (patterns). Then the LBP feature is generated by using the histogram of LBP code map. However, some binary patterns may frequently appeared in the images. In [14], uniform pattern is defined if there are at most two 0-1 or 1-0 transitions. The example showed in Figure 3.7 is a uniform pattern. In total, we can find 58 uniform patterns. Finally, we define a separate bin for each uniform pattern, and all the other patterns are assigned to one bin. Thus, the number of patterns as well as the feature dimensions are reduced from 256 to 59.

Bag-of-Visual-Words (BoVW) [39] borrows the idea of BoW except the words are descriptors which are computed on image patches around some keypoints. The location of keypoints are detected by using the difference of Gaussians function (DoG) applied in scale space. Another method widely used for keypoint detection is the Hessian-Affine region detector. Around the detected keypoint, the dominant orientation and scale is

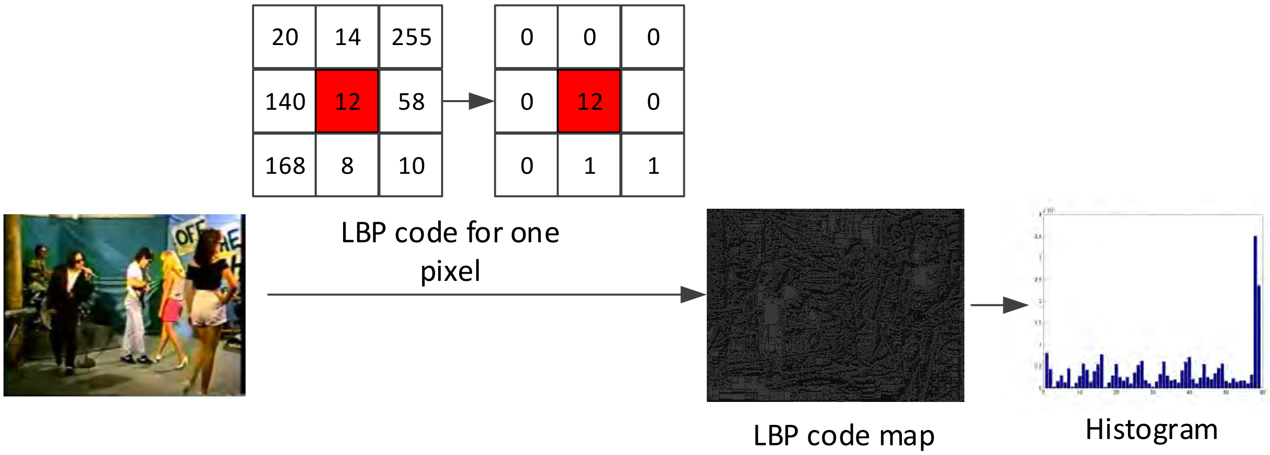
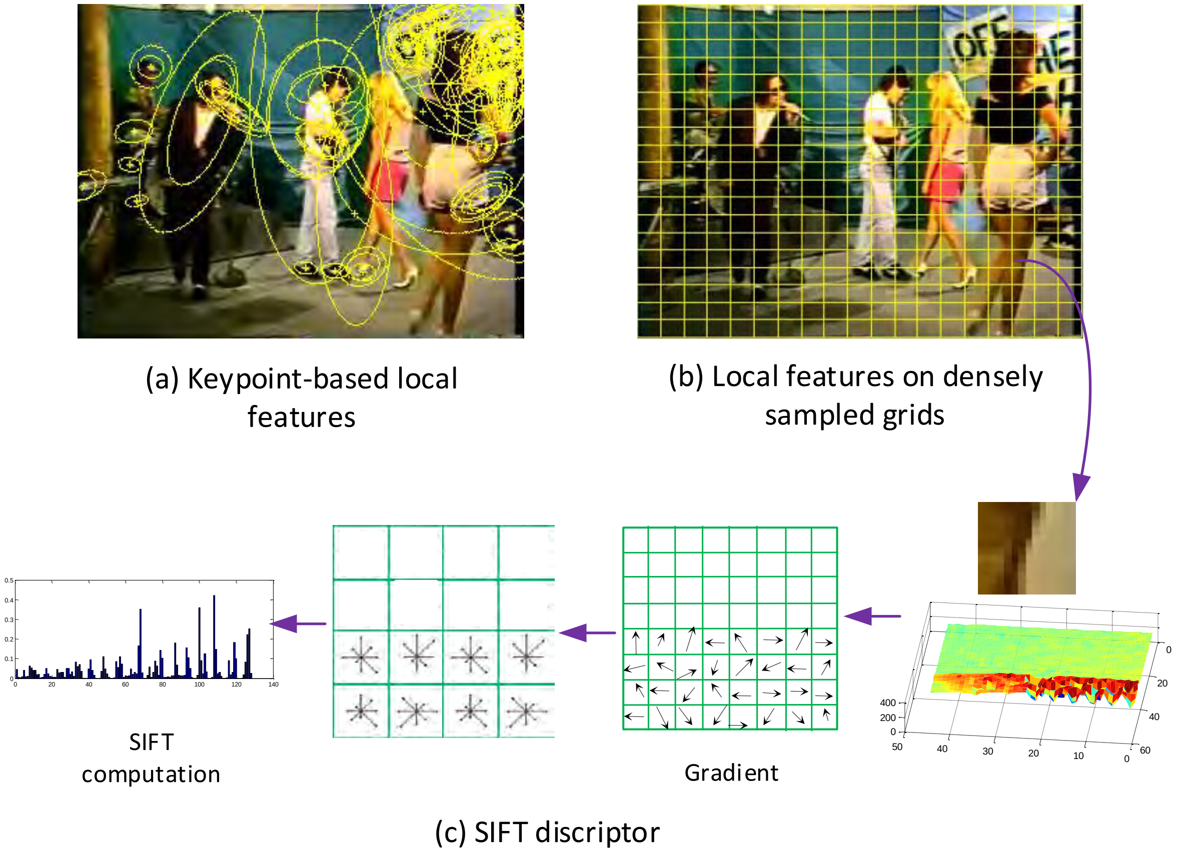


Figure 3.7: The process for extracting LBP feature for a single image.

determined by analyzing the gradient values. An example of the detected keypoints are showed in Figure 3.8(a). In this way, the extracted feature is invariant to translation, scaling, rotation, illumination changes. For visual recognition, instead of extracting local features around keypoints, it has been observed that the method using densely sampled local regions (Figure 3.8(b)) can also achieve comparable results.

After detecting the most informative regions, SIFT descriptor [30] is adopted to represent the selected image patches. As showed in Figure 3.8, the gradient (magnitude and orientation) of each pixel is first computed on the patch scaled to fixed size. Gaussian weighting function is further utilized on the image patch so that the pixel far away from the keypoint center will be assigned with less weight. Then, the patch is split to 4 × 4 grids, where 16 histograms are formed to quantize the orientations into 8 bins. Finally the histograms are merged as a 128 dimensional vector.

With the extracted SIFT descriptors, the next step is to quantize the descriptors into a feature vector using Bag-of-Visual-Words (BoVW) model. Each SIFT descriptor can be treated as a visual word similar to the text word in BoW. The process for generating BoVW



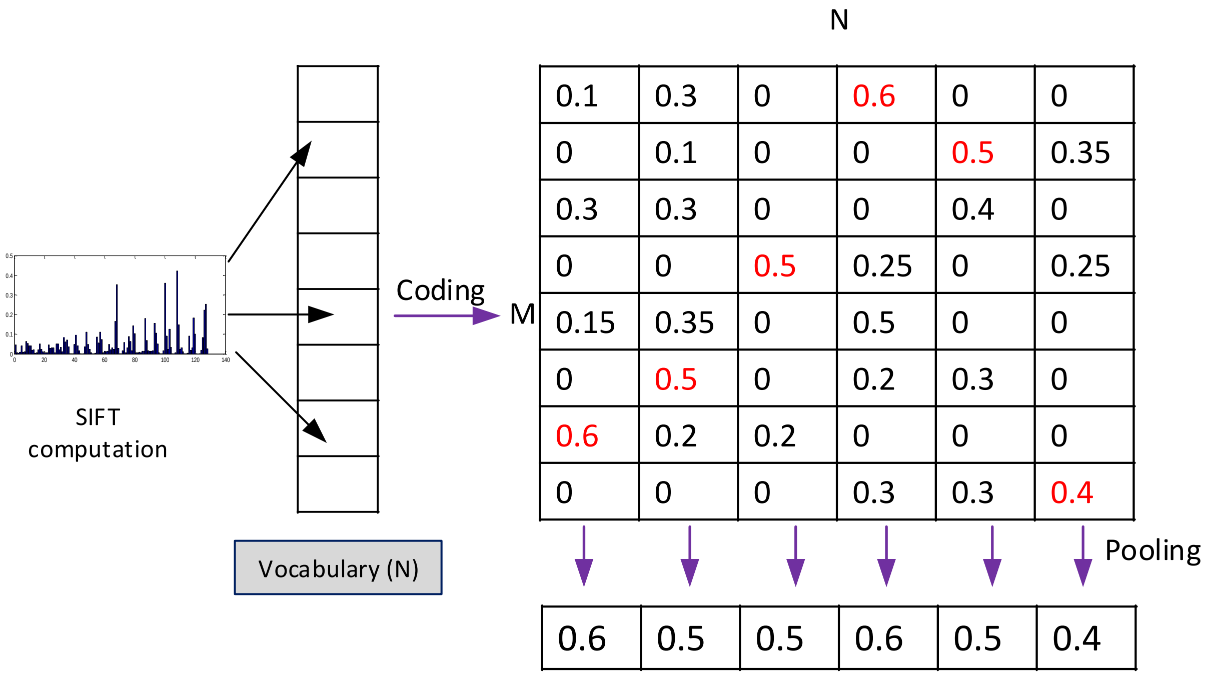
(a) Keypoint-based local features

(b) Local features on densely sampled grids

(c) SIFT discriptor

Figure 3.8: The process for extracting SIFT location feature.

feature is showed in Figure 3.9, where the vocabulary is the centers of groups generated by clustering a set of descriptors extracted from several images. Given an input image, each extracted SIFT descriptor is assigned to one or several visual words in the vocabulary. In the standard hard assignment, the value corresponding to the visual word which is closest to the input SIFT feature will be increased by one. In [21], soft assignment is adopted to allow that each SIFT feature can contribute to K closest visual words with different weights. An advanced method Locality-constrained Linear Coding (LLC) proposed in [68]



0.1	0.3	0	0.6	0	0
0	0.1	0	0	0.5	0.35
0.3	0.3	0	0	0.4	0
0	0	0.5	0.25	0	0.25
0.15	0.35	0	0.5	0	0
0	0.5	0	0.2	0.3	0
0.6	0.2	0.2	0	0	0
0	0	0	0.3	0.3	0.4

0.6	0.5	0.5	0.6	0.5	0.4

Figure 3.9: Bag-of-Visual-Words (BoVW) generated by quantizing SIFT descriptors.

adopts the reconstruction coefficients as weights using the following equation:

$$\begin{aligned} c^* &= \arg\min_c \|x_i - c_i^T B\|^2, \\ &s.t., \Sigma_j c_j = 1, \end{aligned} \tag{3.7}$$

where B is the vocabulary including N words, x_i is the SIFT descriptor and c^* is the learned coefficients. For an image containing M SIFT descriptors, the quantization results will be a $M \times N$ matrix as showed in Figure 3.9. The BoVW feature is generated by pooling along each column. Maximum pooling selects the largest value, while average pooling computes the average value in each column. In this thesis, we will use the maximum pooling strategy which has been demonstrated better than average pooling on image recognition.

3.3.2 Middle-level Visual Features

The middle-level features adopted in this thesis include Classemes, Attribute and SentiANP. We will illustrate each of them.

Classemes [66] is a middle-level feature which adopts the outputs of 2,659 classifiers trained for detecting some semantic concepts (e.g., objects). Each dimension indicates the probability of the appearance of a category. Classemes is built for general category image search. The 2,659 concrete concepts are selected from the Large Scale Concept Ontology for Multimedia (LSCOM). For example, some highly weighted classes are "helmet", "walking" and "body_of_water". Each classifier is learned using LP-β kernel combiner, where 13 types of low-level features are combined using different kernels such as χ^2 distance, i.e., $k(x, y) = exp(-\chi^2(x, y)/\gamma)$. The adopted features are:

- Color GIST descriptor as described in section 3.3.1.
- Pyramid of Histograms of Oriented Gradients (PHOG) [6] and Unoriented Gradients under four spatial pyramid scales.
- Pyramid self-similarity [55] is defined as a 30 dimensional descriptor at every 5 pixels. Quantization is performed in a similar way with BoVW to generate the feature vector with three spatial pyramid levels.
- BoVW using keypoints detected by the Hessian-Affine detector.

Comparing to low-level visual features, Classemes represents images at a higher semantic level. It is expected to be helpful to detect challenging concepts with large within-class variants, which are hard to be captured from low-level features. For example, duck can be inferred from the detection results of concepts in Classemes such as bomber_plane, body_of_water and swimmer.

Attribute is another middle-level feature, which represents abstract visual aspects (adjectives, e.g., "red" and "striped"), rather than the concrete objects used in Classemes. We adopt the 2,000 dimensional attribute proposed in [80], which is designed for representing category-ware attributes. Firstly, a category-attribute matrix A is learned. Each row can be considered as the representation of a category in the attribute space or the associations between the category and attributes. The objective function is defined as

$$\max_{A} J(A) = \sum_{i,j} \parallel A_{i.} - A_{j.} \parallel_2^2 - \lambda \sum_{i,j} S_{i,j} \parallel A_{i.} - A_{j.} \parallel_2^2 \tag{3.8}$$

where the first term induces discriminative capability, the second term is the proximity preserving regularization term, and $S_{i,j}$ measures the category similarity. Then, attribute classifiers are learned using the category labels on training instances weighted by the learned associations A. In this way, the attribute classifiers can be obtained without the requirement of human supervision on the training data. This is significantly important for attribute learning as annotating attributes is much more expensive than general concepts.

SentiANP [3] is an attribute representation dedicatedly designed for human affective computing in Sentibank ontology. It includes 1,200 Adjective Noun Pairs (ANPs), e.g., "cloudy moon" and "beautiful rose" which are carefully selected from Web data. ANP is more detectable than the adjectives only, meanwhile it is representative for expressing human affects. For each ANP, the training instances are collected from Flickr. The classifier learning follows the similar way with other middle-level features except the utilized low-level features. Comparing to Classemes and Attribute, which are designed for general visual understanding, SentiANP is intuitively suitable for visual sentiment analysis.

3.3.3 Aesthetic Features

In this section, we consider to use two kinds of recently proposed features based on the psycho-visual statistics.

Aesthetic (AF) [1] feature is helpful for understanding the visual instance at more abstract level such as "beautiful". It can partially reflect the affections expressed in the visual instance. We adopt following aesthetic features used in [1]:

- Dark channel feature utilizes the minimum filter on RGB channels of each pixel. The image I is first divided into $m \times m$ grids $\{A_j\}_{j=1}^{m\times m}$. The dark channel feature for each grid is defined as

$$F_{dc}(j) = \sum_{i\in A_j} \frac{\min\limits_{c\in R,G,B}(\min\limits_{k\in\mathcal{N}(i)} I_c(k))}{\sum\limits_{c\in R,G,B} I_c(i)}, \tag{3.9}$$

  where $\mathcal{N}(i)$ is the set of neighbors of pixel i. This feature is able to reflect the local clarity and saturation.

- Luminosity feature computed on the luminosity channel in LAB color space is defined as

$$F_{lf}(j) = \exp(\frac{\sum\limits_{i\in A_j} \log(\delta + I_L(i))}{S}), \tag{3.10}$$

  where S is the size of A_j, and I_L is the luminosity channel.

- Sharpness of an image is derived from the spectral map (a_{x_j}) and sharpness map which is computed on the 8-neighbor pixels in A_j. It is defined as

$$F_{S3}(j) = \left[\left(1 - \frac{1}{1+\exp(-3(a_{x_j}-2))}\right)\left(\frac{1}{4}\max_{r,c}\frac{\sum\limits_{r,c}|x_j^r - x_j^c|}{255}\right)\right]^{1/2}, \tag{3.11}$$

  where x_j^r and x_j^c are row and column neighbors of x_j in A_j.

- Symmetry is defined as the difference between left-half and right-half of the image, as well as the difference between top-half and bottom-half. Each half image is first split into several grids, where some low-level features (e.g., color histogram or HoG histogram) are extracted. The difference is computed using the low-level features.

- White balance feature is defined as the difference of an image from the ideal image with adjusted white balance. It can be simply defined as

$$F_{wb} = \frac{\max(ave, 128)}{ave_c}, \tag{3.12}$$

where ave and ave_c are average gray value of three channels and average value of one certain channel in A_j respectively.

- Colorfulness is defined as the number of non-zero elements in the histogram extracted from the hue channel in HSV color space.

- Color harmony is derived from the color patterns appeared in the hue channel of HSV color space. In specific, a harmony value is defined as the intersection between the histogram of the hue channel and a pre-defined color pattern (histogram). Given a set of color patterns, the computed maximum harmony value is used as the aesthetic feature.

- Eye sensitivity measures the sensitivity of an eye to the colors of certain wavelength. The image patch A_j is first represented as a color histogram weighted by the pre-defined color sensitivities. Then the maximum value of the histogram is used as a kind of aesthetic feature.

Principles-of-Art (PoA) features [82] are defined based on some principles of art (e.g., balance and emphasis), rather than the elements of art used in other aesthetic features. PoA has been proved to be helpful in image emotion detection. The defined PoA features include:

- Balance refers to the symmetry of the arrangement of art work. It is similar to the symmetry feature defined in [1], where only horizontal and vertical symmetry are considered. In [82], balance features include bilateral symmetry using feature point matching, rotational symmetry [31] using hough transform and radial symmetry [32] using radial symmetry transformation.

- Emphasis (contrast) indicates sudden and abrupt changes in elements. In [82], Itten's color contrasts [20] and Sun's rate of focused attention (RFA) [62] are employed to represent this property. In [20], the features include contrast of saturation, hue, complements, warmth, cold, harmony and extension. RFA measures the level of attention when viewing the image. With a detected saliency map $Saliency$ and several aesthetic masks $\{Mask_i\}$, RFA is defined as

$$RFA_i = \frac{\sum\limits_{x,y} Saliency(x,y) Mask_i(x,y)}{\sum\limits_{x,y} Saliency(x,y)} \tag{3.13}$$

- Harmony (unity) represents a sense of completion and uniform appearance, and is reflected by repetition or gradual changes of elements. In [25], a harmony score is defined for each pixel using the hue and gradient of its circular neighbors. The neighbors are first split into two groups c_1 and c_2, each of which includes several adjacent neighbors. The harmony score at pixel (x, y) is defined as

$$H(x,y) = \min_{c_1} \exp\left(-|h(c_1) - h(c_2)|\right) |i(c_1) - i(c_2)|, \tag{3.14}$$

  where $h(c_1)$ and $h(c_2)$ are maximum hue or gradient in group c_1 and c_2 respectively, and $|i(c_1) - i(c_2)|$ indicates the shortest circular distance between the neighbors with maximum hue or gradient. The overall harmony is the sum of scores in the image.

- Variety indicates the complexity of elements in the image, which affect the visual interestingness. This feature is defined as the color variety by counting different basic colors, and distribution of gradient on eight directions and eight scales.

- Gradation is the way of change from one element to another element. Pixel-level gradation is defined as windowed total variation (WTV) and windowed inherent variation (WIV) [75]. WTVs in x and y directions at pixel $p(x, y)$ is defined as

$$\begin{aligned} D_x(p) &= \sum_{q \in R(p)} g_{p,q} |(\partial_x I)_q| \\ D_y(p) &= \sum_{q \in R(p)} g_{p,q} |(\partial_y I)_q| \end{aligned} \tag{3.15}$$

where $R(p)$ is a rectangular region around p. $D_x(p)$ and $D_y(p)$ count the absolute spatial difference within $R(p)$, and weighted by

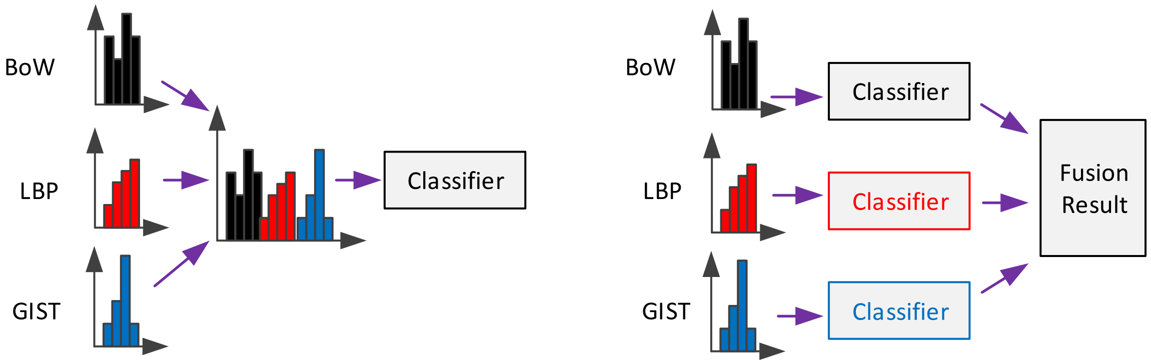
$$g_{p,q} = \exp\left(-\frac{(x_p - x_q)^2 + (y_p - y_q)^2}{2\sigma^2}\right) \tag{3.16}$$

Similarly, we can define WIV as

$$\begin{aligned} L_x(p) &= \sum_{q \in R(p)} |g_{p,q} (\partial_x I)_q| \\ L_y(p) &= \sum_{q \in R(p)} |g_{p,q} (\partial_y I)_q| \end{aligned} \tag{3.17}$$

The relative and absolute gradations are represented by the sum of relative total variation (RTV), WTV and WIV defined as

$$\begin{aligned} RG &= \sum_p RTV(p) = \sum_p \left(\frac{D_x(p)}{L_x(p) + \varepsilon} + \frac{D_y(p)}{L_y(p) + \varepsilon} \right) \\ AGT_x &= \sum_p D_x(p) \\ AGT_y &= \sum_p D_y(p) \\ AGI_x &= \sum_p L_x(p) \\ AGI_y &= \sum_p L_y(p) \end{aligned} \tag{3.18}$$



(a) Early fusion of multiple features

(b) Late fusion of multiple features

Figure 3.10: The widely used fusion strategies: (a) early fusion utilize a single model learned on the feature by concatenating multiple features, (b) late fusion combines the outputs of multiple classifiers learned on different features respectively.

3.4 Multi-view Sentiment Analysis

3.4.1 Fusion Strategies

Multi-view analysis makes use of the information extracted from both textual and visual aspects of a tweet. The most straightforward and standard way is fusing the information from two views using either early fusion or late fusion. We use x_t and x_v to denote the extracted textual and visual features respectively. In early fusion as showed in Figure 3.10(a), the two features extracted from text and image are concatenated into a single feature vector $x = \{x_t, x_v\}$. The model used for predicting sentiment is defined as $Score = f(x)$. In contrast, late fusion showed in Figure 3.10(b) combines the output scores of several models (classifiers) learned on textual and visual data respectively. The final prediction result is $Score = (f(x_t) + f(x_v))/2$ by assigning same weights for different features. Simple fusion

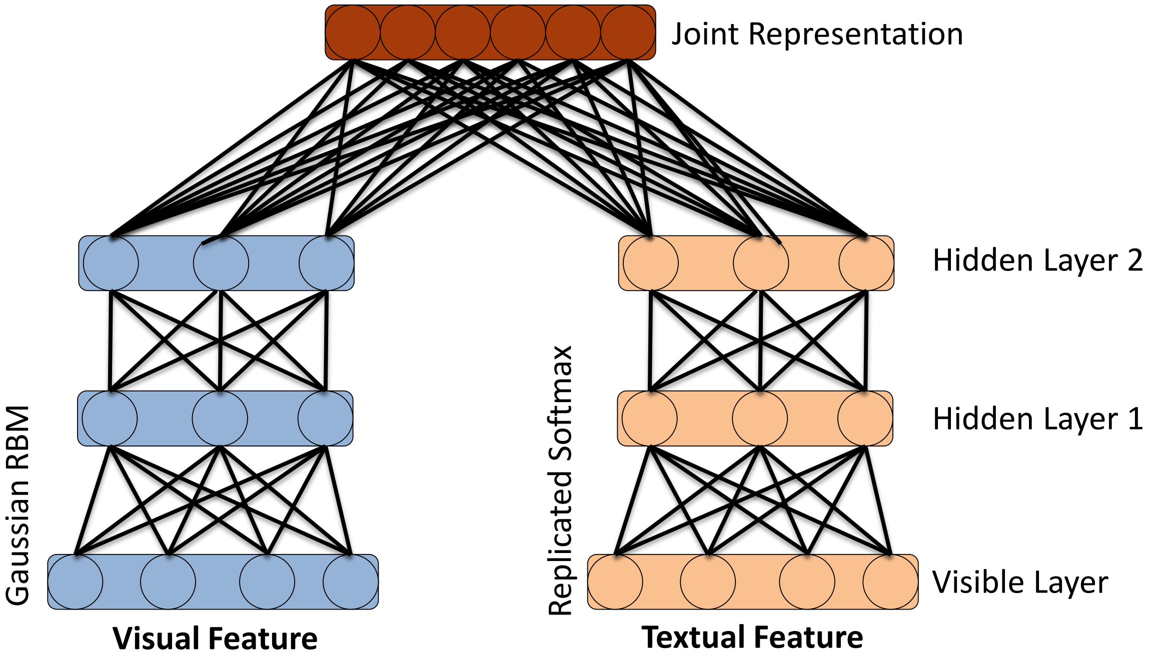


Figure 3.11: The Multi-model Deep Boltzmann machine (DBM) for learning joint representation from both text and visual views.

strategies can be also used for combining different kinds of features from single view. We will also evaluate this in the experiments.

3.4.2 Joint Feature Learning

While both early and late fusion are able to boost the performance, the inherent correlations between two views are not considered. Recently, multi-model learning method has showed strong performance on multi-view data analysis. In this thesis, we adopt the approaches proposed in [60], where a multi-model Deep Boltzmann Machine (DBM) is trained using textual and visual features as inputs. In [45], it has been showed to be helpful in detecting emotions from Web videos. In this work, we use a similar architecture by only using the visual and textual pathways. The input of the visual pathway is a 20,651 dimensional feature combining the Dense SIFT, HOG, SSIM, GIST, and LBP. In the textual pathway, BoW representation is utilized for generating input features. Each pathway is formed by stacking multiple Restricted Boltzmann Machines (RBM). The joint layer upon the two pathways contains 2,048 hidden units. Each RBM models the non-linear relationship between different features, stacking two RBM can model more complicated correlations with two non-linear transformations. On the other hand, greedy learning through two layers improves the feature representation in the way that correlation is learned, and meanwhile important information is not missed.

The visible layers of visual pathway and textual pathway are denoted as V^k and V^t respectively. We denote the first and second hidden layers in visual pathway as $\mathbf{h}^{(1k)} \in \{0,1\}^{F_1^k}$ and $\mathbf{h}^{(2k)} \in \{0,1\}^{F_2^k}$ respectively. Similarly, the two hidden layers in the textual pathway are $\mathbf{h}^{(1t)} \in \{0,1\}^{F_1^t}$ and $\mathbf{h}^{(2t)} \in \{0,1\}^{F_2^t}$. In the visual pathway, the connections between $\mathbf{v}^k$ and $\mathbf{h}^{(1k)}$ are modeled with Gaussian RBM [15] and the connections between $\mathbf{h}^{(1k)}$ and $\mathbf{h}^{(2k)}$ are modeled with standard binary RBM. The probability distribution over $\mathbf{v}^k$ is given by

$$P(\mathbf{v}^k;\theta^k) = \frac{1}{\mathcal{Z}(\theta^k)} \sum_{\mathbf{h}^{(1k)},\mathbf{h}^{(2k)}} exp(-E(\mathbf{v}^k,\mathbf{h}^{(1k)},\mathbf{h}^{(2k)};\theta^k) \tag{3.19}$$

where $\mathcal{Z}(\theta^k)$ is the partition function and the free energy E is defined as

$$\begin{aligned} E(\mathbf{v}^k,\mathbf{h}^{(1k)},\mathbf{h}^{(2k)};\theta^k) = \sum_i \frac{(v_i^k - b_i^k)^2}{2(\delta_i^k)^2} - \sum_{ij} \frac{v_i^k}{\delta_i^k} W_{ij}^{(1k)} h_j^{(1k)} \\ - \sum_{jl} h_j^{(1k)} W_{jl}^{(2k)} h_j^{(2k)} \end{aligned} \tag{3.20}$$

where $\theta^k = \{\mathbf{W}^{(1k)}, \mathbf{W}^{(2k)}\}$ are the model parameters. Different from the visual pathway with real-value inputs, the textual features are discrete values (i.e., count of words). Thus, we use Replicated Softmax [51] to model the distribution. The probability of generating

$\mathbf{v}^t$ is given by

$$P(\mathbf{v}^t;\theta^t) = \frac{1}{\mathcal{Z}(\theta^t)} \sum_{\mathbf{h}^{(1t)},\mathbf{h}^{(2t)}} exp(\sum_{jk} W_{kj}^{(1t)} h_j^{(1t)} v_k^t + \sum_{jl} W_{jl}^{(2t)} h_j^{(1t)} h_l^{(2t)} + N \sum_j b_j^{(1t)} h_j^{(1t)}) \tag{3.21}$$

where $\theta^t = \{\mathbf{W}^{(1t)}, \mathbf{W}^{(2t)}, \mathbf{b}\}$ are model parameters. Finally, a joint layer donated as $\mathbf{h}^J \in \{0,1\}^{F^J}$ is added upon layer $\mathbf{h}^{(2k)}$ and $\mathbf{h}^{(2t)}$. The overall joint density distribution over all the inputs is

$$P(\mathbf{v};\theta) = \sum_{\mathbf{h}^{(2k)},\mathbf{h}^{(2t)},\mathbf{h}^{(J)}} P(\mathbf{h}^{(2k)}, \mathbf{h}^{(2t)}, \mathbf{h}^{(J)}) (\sum_{\mathbf{h}^{(1k)}} P(\mathbf{v}^k, \mathbf{h}^{(1k)} \mid \mathbf{h}^{(2k)})) (\sum_{\mathbf{h}^{(1t)}} P(\mathbf{v}^t, \mathbf{h}^{(1t)} \mid \mathbf{h}^{(2t)})) \tag{3.22}$$

The model learning includes two steps. Firstly, each RBM is pre-trained using the greedy layerwise pre-training strategy [50]. Then, the learned parameters are used as initialized parameters for fine-tuning the multimodal DBM in a unified way. According to [16], 1-step contrastive divergence (CD$_1$) is adopted for pre-training, and persistent contrastive divergence (PCD) is utilized for the fine-turning. In this thesis, we utilize the code provided in [60][3] for the learning and inference. The data used in this model learning is the 827,659 text-image pairs provided by SentiBank dataset [3].

3.5 Classifier Learning

Sentiment analysis is intuitively a multi-class classification problem. The widely used ways for simplifying multi-class classification to a set of binary classification sub-problems are one-vs-one and one-vs-rest strategy. In one-vs-one method, a classifier is built for splitting two classes in the datasets. In one-vs-rest method, we built one classifier for each

[3] http://www.cs.toronto.edu/∼nitish/multimodal/

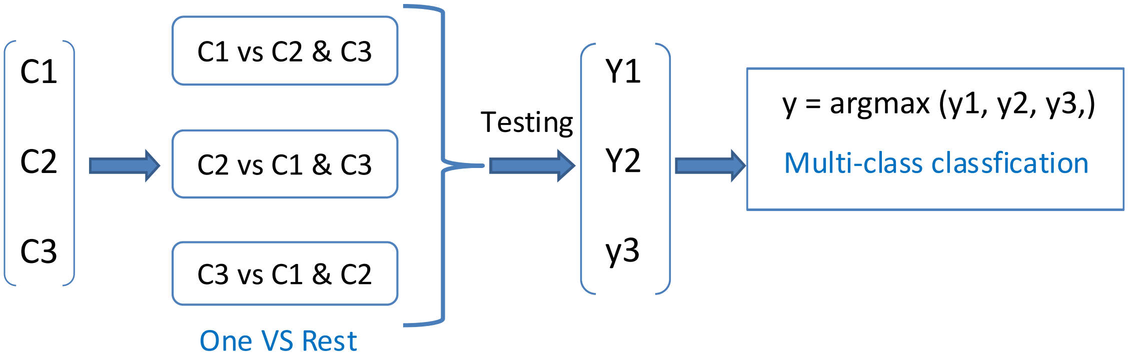


Figure 3.12: Classifier training and testing using one-vs-rest strategy.

category to differentiate the target class (positive class) with all other classes (negative class). As showed in Figure 3.12, three classifiers are needed for three classes. These two strategies archive similar performance. However, with the increasing number of classes, much less classifiers are needed in one-vs-rest strategy. In this thesis, the number of classes is usually two (positive and negative) or three (positive, neutral and negative). In case of two classes, one binary classifier is enough. Otherwise, we can adopt the one-vs-rest strategy. Given a testing instance, the learned classifiers are employed on the corresponding extracted feature. The testing instance is eventually assigned to the class with the maximum prediction score.

In this thesis, we fix the statistical learning method as linear SVM, which has showed robust performance on different kinds of recognition tasks involving various media types. There are many hyperplanes that can successfully separate two classes. In SVM, the hyperplane is defined as the one which can archive largest separation or margin between two classes. As showed in Figure 3.13, the aim is to search a hyperplane so that the distance from it to the nearest instance can be maximized. Meanwhile, the instances can be separated in the right side. This can be defined as

$$\begin{aligned} &\arg\min_{W,b} \frac{1}{2}\|W\|^2 \\ &\text{subject to} \\ &y_i(Wx_i - b) \geq 1, i \in [1, n] \end{aligned} \tag{3.23}$$

where x_i and $y_i \in \{+1, -1\}$ are feature and label for instance i respectively. As showed in Figure 3.13, there are no instances filling in the margin. This may be too rigid in some applications. So soft margin SVM is proposed to allow few instances can be misclassified within the margin by introducing non-negative slake variables ξ_i. The training instances with non-zero ξ_i will be penalized. The optimization function becomes

$$\begin{aligned} &\arg\min_{W,b,\xi} \frac{1}{2}\|W\|^2 + C\sum_{i=1}^{n}\xi_i \\ &\text{subject to} \\ &y_i(Wx_i - b) \geq 1 - \xi_i, \xi_i \geq 0, i \in [1, n] \end{aligned} \tag{3.24}$$

where C controls the trade-off between the large margin and the classification error. Finally, the solution of W is the weighted linear combination of training instances on or within the margin. These training instances are called support vectors.

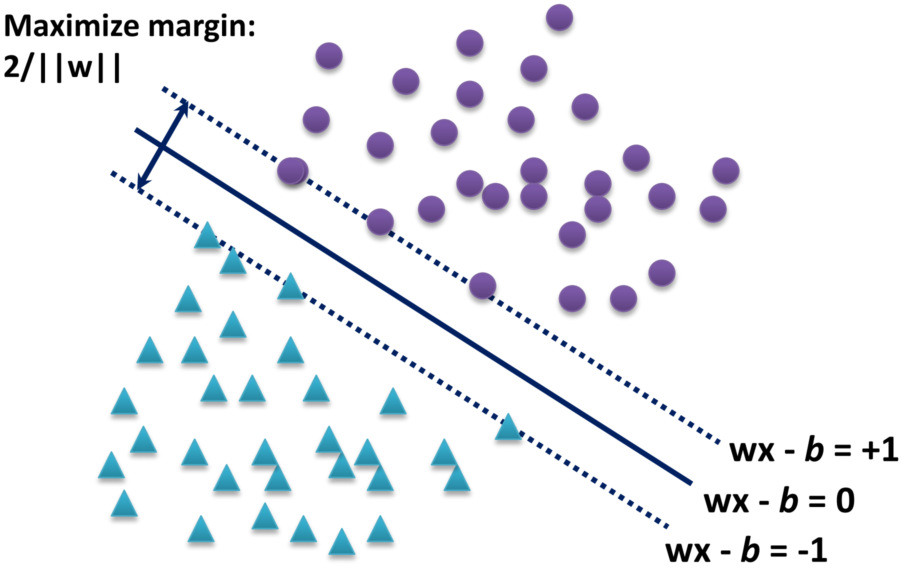


Figure 3.13: Maximum margin hyperplane for separating two classes in SVM.

Chapter 4

The MVSA Dataset

4.1 Data Collection

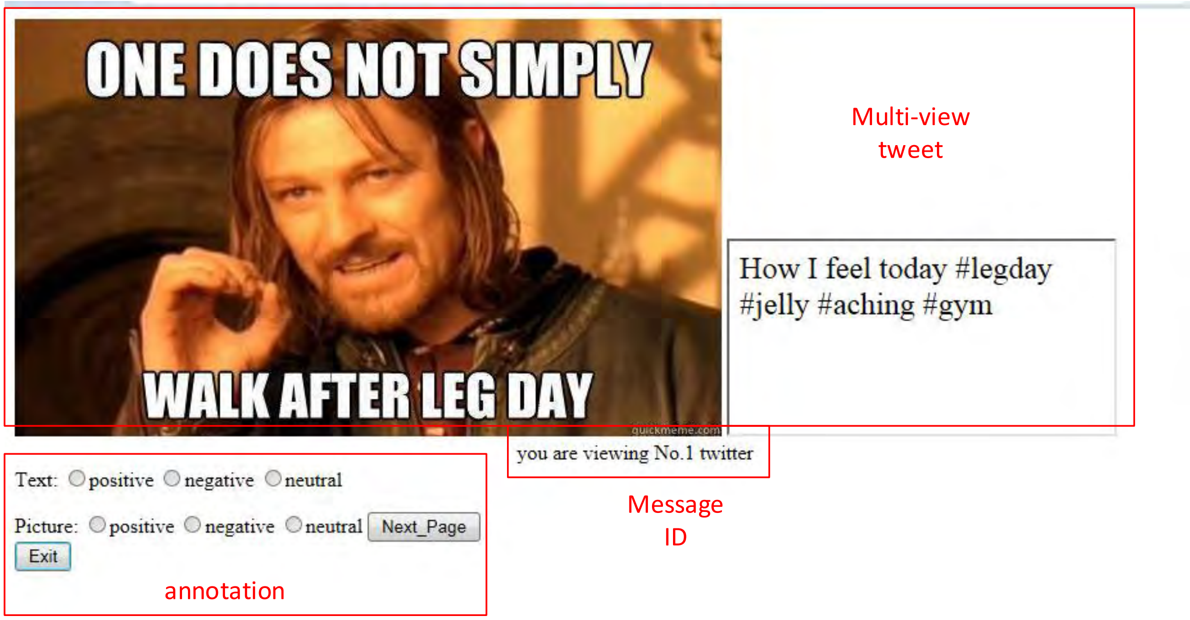
All the image-text pairs in MVSA are collected from instant message sharing website Twitter, which has over 300 million active users and includes 500 million new tweets per day[1]. We adopt a public streaming Twitter API (Twitter4J)[2] for collecting data.

In order to collect representative tweets from the large volume of data, the twitter stream is filtered by using a vocabulary of 406 emotional words[3]. In specific, only the tweets containing keywords in the messages or hashtags are downloaded. Since many users are lazy to write a lot, or the things are complicated to describe, Hashtag can be a convenient and accurate way for users to express and emphasize their interested contents or opinions. The vocabulary used for filtering the messages includes ten distinct categories (e.g., happiness, caring and depression) covering almost all the sentiments of human beings. In each category, keywords are grouped into three degrees (strong, medium, light) which can measure their levels of emotion. Some emotional words, such as happy and sad, frequently

[1] https://about.twitter.com/company

[2] http://twitter4j.org/en/

[3] http://www.sba.pdx.edu/faculty/mblake/448/FeelingsList.pdf



How I feel today #legday #jelly #aching #gym

you are viewing No.1 twitter

Text: ○positive ○negative ○neutral

Picture: ○positive ○negative ○neutral

Multi-view tweet

Message ID

annotation

Figure 4.1: Annotation interface.

appear in the tweets. To balance the collected data among different emotions, we used the keywords roundly and collected at most 100 tweets for one keyword at each round. In addition, the data collection was daily performed at several time slots during one day, thus the contents can be diversified. After downloading the tweets, we will extract the image URLs within the messages to further download the paired images. Only the text-image tweets with accessible images are kept for annotation.

4.2 Annotation Process

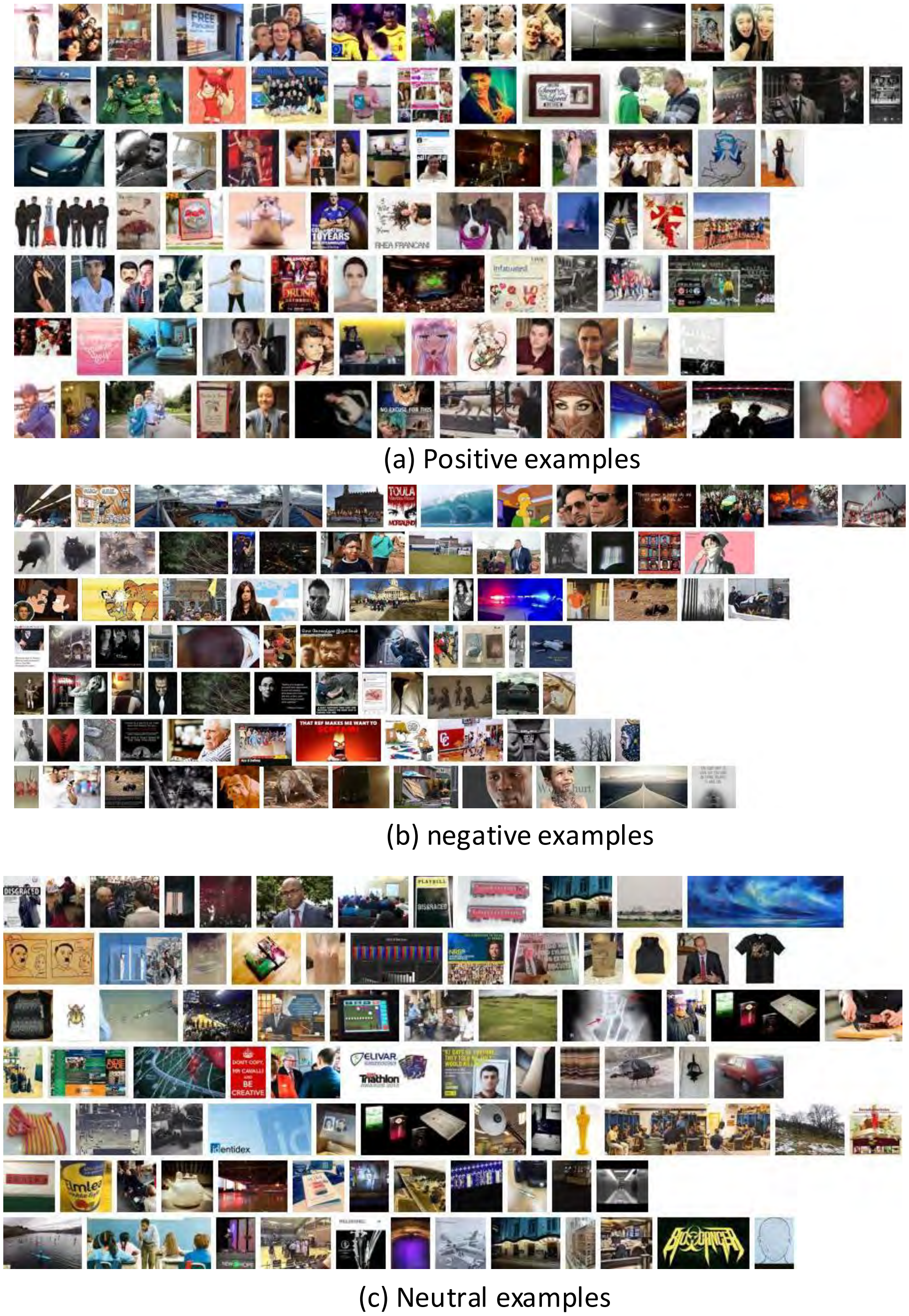
Annotating sentiments on large set of image-text pairs is difficult, particularly when uncontrolled Web users may post messages without correlations between the image and text. To facilitate the annotation, we developed an interface showed in Figure 4.1. Every user

Table 4.1: Statistics of manually annotated datasets for tweet sentiment analysis.

Dataset	#Positive	#Negative	#Neutral	Data type
HCR	541	1,381	470	text
STS	182	177	139	text
SemEval	5,349	2,186	6,440	text
STS-Gold	632	1,402	77	text
Sanders	570	654	2,503	text
ImgTweet	769	500	-	image
Sentibank	470	133	-	text+image
MVSA	1398	724	470	text+image

needs a unique ID for accessing this interface. Each time, an image-text pair is shown to an annotator, who will assign one of the three sentiments (positive, negative and neutral) to the text and image separately. To ensure effective annotations, the button of next message is valid only when the labels for two views are received. If two labels are the same, both the two views are very likely to reflect a same human affection. Note that text and image in a message do not necessary have a same sentiment label. The annotations can be used for generating three subsets of data corresponding to text, image and multi-view respectively.

Until now, the dataset has received annotations for 4,869 messages. We only include the tweets that receive same labels on text and image as the final benchmark dataset. Table 4.1 lists the details of MVSA and several popular public datasets for sentiment analysis of tweets. Comparing to other datasets, MVSA is already the largest dataset for multi-view sentiment analysis. Figure 4.2 shows some examples in three sentiment categories. We will keep increasing the dataset by including more up-to-date messages, and the annotations will be regularly released.

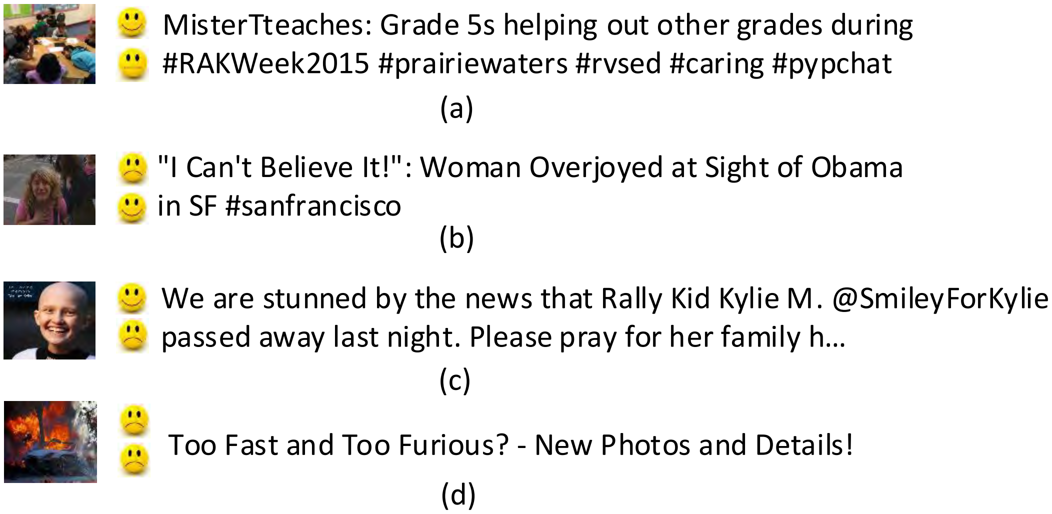


(a) Positive examples

(b) negative examples

(c) Neutral examples

Figure 4.2: Examples of positive, negative and neutral images in our MVSA dataset.


🙂 🙂 MisterTteaches: Grade 5s helping out other grades during #RAKWeek2015 #prairiewaters #rvsed #caring #pypchat

(a)

🙁 🙂 "I Can't Believe It!": Woman Overjoyed at Sight of Obama in SF #sanfrancisco

(b)

🙂 🙁 We are stunned by the news that Rally Kid Kylie M. @SmileyForKylie passed away last night. Please pray for her family h...

(c)

🙁 🙁 Too Fast and Too Furious? - New Photos and Details!

(d)

Figure 4.3: Example tweets with both image and text. The top and bottom icons in the middle indicate sentiments (positive, negative or neutral) showed in image and text respectively.

4.3 Data Analysis

We have observed that there are inconsistent sentiments represented in user posted image and the corresponding text. This is because that the motivations of posting both text and image may not be always to enhance the sentiment or emotion of users. For example, text showed in Figure 4.3(a) is the description of the event in the photo. The two views are visually related, rather than emotionally related. Another reason is that sentiment expressed in the message is usually affected by the contexts of posting this message. For example, Figure 4.3(b) shows a crying woman in the picture, while textual part indicates that the woman was overjoyed at seeing Obama. In contrast, Figure 4.3(c) is a photo of a smiling kid, however, the fact is that the kid passed away as described in the text. Besides these, image and text can enhance the users' sentiment. In Figure 4.3(d), there is a weak negative sentiment in the text, which is strengthened by the attached image about a firing

Table 4.2: The percentage of messages with same labels in both textual and visual views. The messages are grouped into 10 categories based on the contained emotional keywords.

Category	Anger	Caring	Confusion	Depression	Fear	Happiness
Agreement(%)	47.5	64.6	47.5	48.7	50.9	64.6

Category	Hurt	Inadequateness	Loneliness	Remorse	**Overall**
Agreement(%)	48.4	46.1	49.1	51.0	**53.2**

Table 4.3: Performance of sentiment analysis using keyword matching method. The 406 emotional keywords are grouped into 10 categories.

Category	Anger	Caring	Confusion	Depression	Fear	Happiness
Accuracy(%)	19.4	50.7	12.3	20.7	18.6	51.4

Category	Hurt	Inadequateness	Loneliness	Remorse	**Overall**
Accuracy(%)	25.5	15.2	20.6	19.9	**30.6**

car in an accident.

In Table 4.2, we list the percentage of agreements on the labels of text and image in a tweet. The messages are grouped into ten categories based on the contained emotional words. We can see that only 53.2% messages show same sentiments in their posted image and text. Thus the sentiment analysis on multi-view messages is extremely challenging. In addition, users express their feeling about “happiness” and “caring” (agreement of 64.6%) more explicitly using delightful words and images.

We further analyze the accuracy of emotional words for indicating the overall sentiment of a Twitter message. We first manually label the 406 emotional keywords as positive, negative and neutral. The sentiment polarity of each tweet is same as that of the contained emotional keyword. Table 4.3 shows the results grouped by the 10 categories. We can see that the overall accuracy is only 30.6%. The performance is especially poor for category “Confusion”, where keywords are ambiguous for expressing human affection. Again, “happiness” and “caring” perform much better than other categories. Thus, more advanced technique is needed in sentiment analysis.

Chapter 5

Experimental Results

5.1 Experiment Settings and Evaluation Criteria

The adopted linear SVM is learned using the liblinear toolbox [9]. Since different datasets may have different numbers of categories, we fix the sentiment labels as positive or negative for all the datasets. In other words, we perform experiments on the polarity classification of sentiment. For the datasets containing more than two labels, we will only utilize the subset of data labeled as positive and negative in our experiments. In addition, some of the datasets do not provide a split between training and testing sets. We randomly split them into training and testing sets by 50%-50%.

As a binary classification problem, we use accuracy, F-score and average precision (AP) to evaluate the performance. Formally, accuracy is defined as

$$accuracy = \frac{tp + tn}{tp + tn + fp + fn} \tag{5.1}$$

where tp, tn, fp and fn indicate true positive, true negative, false positive and false negative respectively.

F-score is defined as

$$F\text{-}score = \frac{2 \times precision \times recall}{precision + recall} \tag{5.2}$$

where precision and recall are calculated by

$$\begin{aligned} precision &= \frac{tp}{tp + fp} \\ recall &= \frac{tp}{tp + fn} \end{aligned} \tag{5.3}$$

In our experiment, F-score is computed on positive class (F-positive) and negative class (F-negative) respectively, and their average is denoted as F-average.

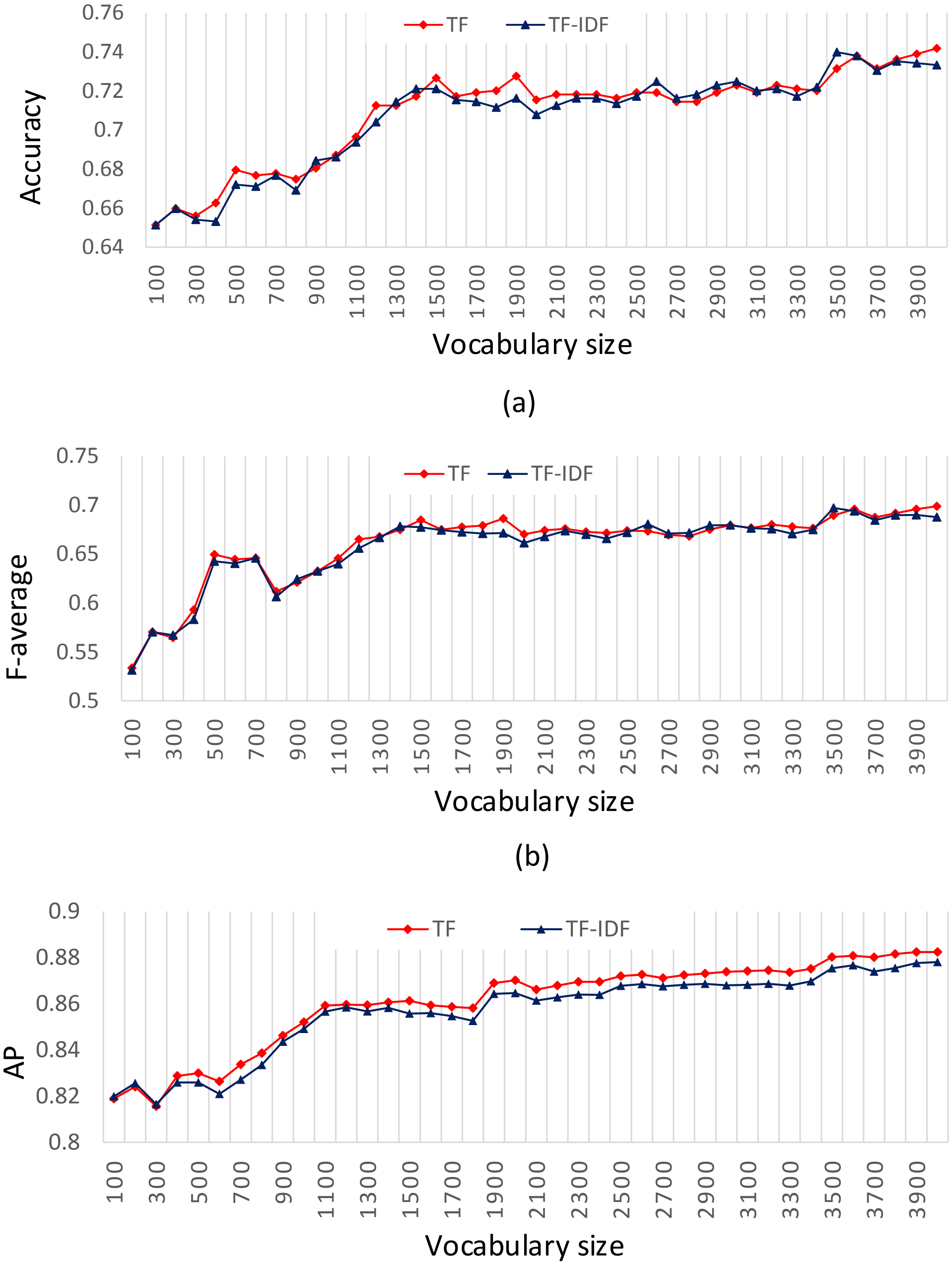
AP is usually used for evaluating the quality of ranked lists such as the search results in information retrieval. In our experiment, AP can also be used for evaluating the classification results. We assume positive sentiment to be the positive instance in classifier learning. All the testing instances are first ranked based on their prediction scores. AP can be defined as

$$AP = \frac{1}{R} \sum_{n=1}^{N} I_n \times \frac{R_n}{n} \tag{5.4}$$

where $I_n = 1$ if the item ranked at n^{th} position is positive sentiment, and $I_n = 0$ otherwise. R is the number of positive items, and R_n is the number of positive items for the top-n items. In this way, $AP = 1$ if all the positive instances receive higher scores than negative instances, which is the ideal result.

5.2 Results on Textual messages

In this section, we will discuss the performance of text sentiment analysis. Several influential aspects in the BoW feature extraction will be discussed first. Then a performance comparison for different approaches will be given.



(a)

(b)

(c)

Figure 5.1: Performance of BoW (TF and TF-IDF) on MVSA dataset using different vocabulary sizes evaluated by (a) accuracy, (b) F-average and (c) AP respectively.

Table 5.1: Prediction results on Sanders testing set. Each value in the table is the accuracy of classifier learned using training instances from the row dataset, and the vocabulary in the BoW feature extraction is generated from the column dataset.

	Sanders	SemEval	STS	Sentibank	STS-Gold
Sanders	**0.735**	0.467	0.559	0.449	0.515
SemEval	0.634	0.537	0.517	0.469	0.491
STS	0.647	0.462	0.515	0.500	0.559
Sentibank	0.482	0.464	0.488	0.471	0.466
STS-Gold	0.612	0.497	0.552	0.506	0.554

5.2.1 Effect of Vocabulary Size

We first evaluate the impact of vocabulary size in BoW feature extraction. The experiments are conducted on our constructed dataset MVSA. Figure 5.1 shows the performance of TF and TF-IDF strategies with various vocabulary sizes. The overall trend is that the performance can be improved using larger vocabulary size with respect to accuracy, F-average and AP. Basically, there is no much difference between TFand TF-IDF. TF-IDF performs even worse than TF for larger vocabulary size with respect to AP in Figure 5.1(c). This is different from the conclusion in general document classification. The reason may be that the IDF assigns large weights to rare words, rather than words reflecting human feelings. For example, “path” is assigned with a larger weight than “good”. This may pose negative impact on the sentiment analysis. We observe that the performance becomes stable when the vocabulary size reaches 2,000. In the following, we fix the vocabulary size to be 2,000 for the BoW feature.

Table 5.2: Prediction results on SemEval testing set. Each value in the table is the accuracy of classifier learned using training instances from the row dataset, and the vocabulary in the BoW feature extraction is generated from the column dataset.

	Sanders	SemEval	STS	Sentibank	STS-Gold
Sanders	0.470	0.506	0.506	0.415	0.515
SemEval	0.571	**0.766**	0.557	0.579	0.550
STS	0.488	0.617	0.535	0.553	0.436
Sentibank	0.736	0.732	0.730	0.738	0.724
STS-Gold	0.319	0.353	0.310	0.306	0.336

5.2.2 Effect of Domain Shift

Another important aspect in all the classification problems is the domain shift caused by the different data distributions between the training and testing sets. Besides this, there is another issue related to the domain shift in text sentiment analysis. The vocabulary in the BoW feature extraction may be generated from a domain different from the applied domain. To evaluate above two issues, we provide a comprehensive study on five textual datasets (i.e., Sanders, SemEval, STS, Sentibank and STS-Gold) listed in Table 4.1.

We first generate five kinds of BoW feature (TF) using different vocabularies generated from the five datasets respectively. For each dataset, there are five classifiers learned on the training set corresponding to the five TF features. Then each of the classifiers is conducted on the five testing sets respectively. We group the results according to the applied testing sets. The accuracy is showed in Table 5.1 to Table 5.5, each of which includes the performance of different classifiers on a testing set. For example, Table 5.1 is the prediction results on the Sanders testing set using different classifiers. Each value in the table indicates the accuracy of the classifier using training instances from the row dataset and the vocabulary is generated from the column dataset. The best result is marked in

Table 5.3: Prediction results on STS testing set. Each value in the table is the accuracy of classifier learned using training instances from the row dataset, and the vocabulary in the BoW feature extraction is generated from the column dataset.

	Sanders	SemEval	STS	Sentibank	STS-Gold
Sanders	0.512	0.540	0.682	0.543	0.467
SemEval	0.537	0.470	**0.746**	0.562	0.554
STS	0.532	0.506	<u>0.713</u>	0.493	0.431
Sentibank	0.504	0.493	0.518	0.479	0.506
STS-Gold	0.493	0.498	0.640	0.487	0.481

Table 5.4: Prediction results on Sentibank testing set. Each value in the table is the accuracy of classifier learned using training instances from the row dataset, and the vocabulary in the BoW feature extraction is generated from the column dataset.

	Sanders	SemEval	STS	Sentibank	STS-Gold
Sanders	0.481	0.418	0.445	0.428	0.495
SemEval	0.601	0.614	0.544	0.584	0.561
STS	0.465	0.518	0.521	0.591	0.508
Sentibank	**0.767**	0.760	0.757	<u>0.738</u>	0.760
STS-Gold	0.239	0.249	0.262	0.302	0.322

bold, and the result of classification without domain shift problem is underlined, which adopts a training set and vocabulary from a domain same with the testing set. We can see that the classifier using within domain vocabulary and training data achieves or approaches the best accuracy for all the five testing sets. For other classifiers with the domain shift problem, the performance degrades significantly. In other words, the domain shift in the BoW feature extraction and training data imposes negative impacts on the performance.

By comparing the accuracy along each row, where classifiers are learned on same train-

Table 5.5: Prediction results on STS-Gold testing set. Each value in the table is the accuracy of classifier learned using training instances from the row dataset, and the vocabulary in the BoW feature extraction is generated from the column dataset.

	Sanders	SemEval	STS	Sentibank	STS-Gold
Sanders	0.522	0.531	0.527	0.554	0.661
SemEval	0.466	0.411	0.399	0.411	0.668
STS	0.516	0.443	0.463	0.457	0.765
Sentibank	0.312	0.312	0.326	0.333	0.326
STS-Gold	0.638	0.638	0.625	0.637	**0.776**

ing instances, we can make a consistent observation that within domain vocabulary always perform better than other vocabularies. For example, for the prediction results on Sanders in Table 5.1, the best result along each row is the one using the vocabulary from Sanders. Similarly, in Table 5.2, where the prediction is performed on the SemEval dataset, the best one in each row adopts the vocabulary from SemEval.

The effect of training data can be observed by comparing the accuracy within each column, where the same vocabulary is used. We can see that using within domain training data cannot guarantee the best column-wise result on the Sanders (Table 5.1), SemEval (Table 5.2) and STS (Table 5.3) datasets. This is because that there are other influential factors in the training set such as the number of instances or distribution between positive and negative instances. For example, SemEval and Sentibank have similar data distributions. Thus, considering the cross-domain performance in Table 5.2, classifiers learned from the Sentibank training data perform better than others on the SemEval testing set. However, we can still observe that within domain training data always perform better than others on Sentibank (Table 5.4) and STS-Gold (Table 5.5).

We further evaluate the cross dataset performance using the SoT feature. The results are showed in Table 5.6, where each value represents the accuracy on the column dataset

Table 5.6: Cross-dataset validation using SoT feature. Each value in the table is the accuracy of classifier learned using training instances from the row dataset, and predicted on column dataset.

	Sanders	SemEval	STS	Sentibank	STS-Gold
Sanders	**0.713**	0.708	**0.774**	0.651	0.764
SemEval	0.563	**0.752**	0.621	0.740	0.499
STS	0.700	0.687	0.763	0.631	**0.776**
Sentibank	0.475	0.748	0.506	**0.780**	0.310
STS-Gold	0.532	0.270	0.493	0.222	0.696

using classifier trained on the row dataset. The best result in each column is marked in bold, and the result without the domain shift problem is underlined. By comparing the results along each column, we can see that best performance is achieved when training and testing instances are from the same dataset for most of the datasets.

In short, the performance of text sentiment analysis is significantly affected by the domain shift in both feature extraction and training data selection. In practice, utilizing data covering diverse domains is expected to be less biased and thus more robust. In the following, we will adopt the vocabulary from the STS training set which is the largest one in our selected datasets.

5.2.3 Performance Comparison on MVSA dataset

Table 5.7 lists the results of different approaches on our constructed MVSA dataset. The best results are highlighted. Note that SentiWordnet and SentiStrength are not suitable to be evaluated by AP.

Generally, statistical learning approaches perform better than lexicon-based approaches. This indicates that sentiment analysis on tweets needs more advanced approaches and ded-

Table 5.7: Performance of different approaches for text sentiment analysis on MVSA dataset.

	F-positive	F-negative	F-average	AP	Accuracy
SentiWordnet	0.640	0.557	0.598	-	0.603
SentiStrength	0.628	0.636	0.632	-	0.632
TF	0.792	0.547	0.670	0.870	0.715
TF-IDF	0.786	0.535	0.661	0.864	0.707
SoT	0.840	0.596	0.718	0.898	0.771
TF+SoT_Early	0.831	**0.648**	0.740	0.913	0.782
TF+SoT_Late	**0.844**	0.643	**0.743**	**0.920**	**0.783**

icated designs. However, there are many factors which may influence the performance of statistical learning approaches, such as the imbalance between positive and negative data. Our dataset includes more positive tweets. As a result, performance of negative class (F-negative) is relatively worse than that of positive class (F-positive) for TF, TF-IDF and SoT. This is consistent with the observation in [49]. In contrast, lexicon-based approaches are employed on each tweet independently. In some cases, it may be helpful to boost the performance of the rare class. Thus, the F-negative of SentiStrength is better than that of the others. In addition, the SoT feature performs much better than the traditional textual features. This indicates that informative signals embedded in the messages are helpful for identifying users' sentiments. Another reason may be that our dataset is constructed by collecting tweets containing emotional keywords, which can be considered as the most informative signal for sentiment analysis. Thus the SoT feature that includes statistics on emotional words is more indicative. We can see that the performance can be further boosted by combining the TF and SoT features using early or late fusion strategies

We further employ the models learned fron different datasets in section 5.2.2 on our dataset. Table 5.8 lists the results using the TF feature. Again, the MVSA classifier

Table 5.8: Prediction results on MVSA dataset using models learned from different datasets with the TF feature.

	F-positive	F-negative	F-average	AP	Accuracy
MVSA	**0.792**	0.547	**0.670**	**0.870**	**0.715**
Sanders	0.157	0.193	0.175	0.770	0.582
SemEval	0.494	0.147	0.321	0.857	0.697
STS	0.752	0.563	0.658	0.817	0.684
Sentibank	0.776	0.054	0.415	0.640	0.639
STS-Gold	0.450	**0.572**	0.511	0.834	0.519

without the domain shift problem performs best. We can see that the F-score of classifiers learned from other datasets varies a lot. This is because that F-score is affected by the adopted threshold for differentiating positive and negative testing instances. In practice, this threshold is difficult to define. We fix this threshold as 0.5 which is the theoretical and default setting. Without considering this factor, we can see that SemEval, which includes more training instances, performs better than other cross-domain classifiers with respect to the AP and Accuracy. Thus sufficient training data is significantly important in sentiment analysis.

5.3 Results on Images

We evaluate the different approaches of image sentiment analysis on ImgTweet, Sentibank and our constructed MVSA datasets.

Table 5.9: Performance of different visual features for sentiment analysis on MVSA dataset. The best result in each kind of visual features is underlined, and the best one of all the approaches is marked in bold.

Feature		F-positive	F-negative	F-average	AP	Accuracy
Low-Level	Color Histogram	0.772	0.263	0.517	0.751	0.652
	GIST	0.777	0.146	0.462	0.681	0.647
	LBP	0.787	0.065	0.426	0.710	0.653
	BoVW	0.775	0.354	0.565	0.740	0.667
Middle-Level	Classemes	0.747	0.431	0.589	0.765	0.650
	Attribute	0.782	0.400	0.591	0.773	0.680
	SentiANP	0.790	0.378	0.584	0.779	0.687
Aesthetic	AF	0.785	0.181	0.483	0.755	0.659
	PoA	0.789	0.051	0.420	0.682	0.655
Fusion	LV-Early	0.780	0.366	0.573	0.770	0.674
	LV-Late	0.788	0.338	0.563	0.774	0.679
	MV-Early	0.776	**0.458**	0.617	0.810	0.683
	MV-Late	0.792	0.392	0.592	0.803	**0.691**
	AV-Early	0.779	0.185	0.482	0.750	0.653
	AV-Late	0.792	0.032	0.412	0.756	0.658
	V-Early	0.781	0.420	**0.600**	**0.812**	0.682
	V-Late	**0.800**	0.149	0.474	0.807	0.676

5.3.1 Within Dataset Performance

In this section, we first perform sentiment analysis within different datasets. The training and testing sets are derived from the same dataset. In addition to the visual features introduced in section 3.3, we further test the early and late fusion of different visual features. Table 5.9, Table 5.10 and Table 5.11 show the results on the MVSA, ImgTweet and Sentibank datasets respectively. Methods starting with LV-, MV-, AV- and V- denote the results of fusing low-level, middle-level, aesthetic and all visual features respectively. For low-level features, we can see that BoVW, which has been proved to be powerful in general image recognition, consistently performs better than the others on the three evaluated datasets. However, BoVW only slightly outperforms the others. This is because visual appearances in a sentiment class are extremely diverse. Local features such as SIFT in BoVW may be not representative for sentiment analysis. There is no obvious winner in the three middle-level features. SentiANP performs relatively better on the MVSA and Sentibank datasets, while Classemes achieves better performance on the ImgTweet dataset. The reason may be that the Sentibank dataset and the MVSA dataset include many human faces, while the images in ImgTweet are more diverse. Classemes is built with many different concept detectors, and thus is more effective on ImgTweet. For aesthetic features, the overall performance of AF is better than PoA on the three datasets. Comparing the three different kinds of features, we can see that middle-level features are more robust than low-level and aesthetic features. This indicates that the middle-level features are not only able to bridge the semantic gap, but also are helpful for narrowing down the affective gap. Aesthetic features are defined according to the high-level human perception on aesthetics and arts which are related to human affection. Thus the performance is better than some popular low-level features. However, user generated images are more diverse, and therefore it performs worse than middle-level features which cover different visual semantics.

Generally, combining the different features, which can capture different visual aspects of an image, can further improve the results over each individual feature. We can see in the table that most of the best results lie in the fields of fusion approaches. However, the fusion of all the features by V-Early and V-Late may fail to boost the performance, as the result is dominated by some robust features (e.g., "SentiANP"). In some cases, the performance may be degraded by some poor features, such as the aesthetic features on the

Table 5.10: Performance of different visual features for sentiment analysis on ImgTweet dataset. The best result in each kind of visual features is underlined, and the best one of all the approaches is marked in bold.

Feature		F-positive	F-negative	F-average	AP	Accuracy
Low-Level	Color Histogram	0.705	0.326	0.515	0.676	0.589
	GIST	0.713	0.468	0.591	0.702	0.627
	LBP	0.732	0.346	0.539	0.704	0.619
	BoVW	0.731	0.505	0.618	0.734	0.651
Middle-Level	Classemes	0.761	**0.603**	0.682	0.790	0.701
	Attribute	0.752	0.545	0.649	0.754	0.679
	SentiANP	0.774	0.581	0.677	0.784	0.706
Aesthetic	AF	0.706	0.335	0.520	0.699	0.593
	PoA	0.741	0.130	0.435	0.621	0.600
Fusion	LV-Early	0.771	0.576	0.674	0.755	0.703
	LV-Late	0.769	0.421	0.595	0.751	0.670
	MV-Early	**0.779**	0.601	**0.690**	**0.812**	**0.716**
	MV-Late	0.771	0.576	0.674	0.801	0.703
	AV-Early	0.702	0.313	0.508	0.678	0.585
	AV-Late	0.731	0.189	0.460	0.684	0.596
	V-Early	0.755	0.560	0.658	0.794	0.686
	V-Late	0.773	0.452	0.613	0.791	0.679

ImgTweet dataset. Furthermore, there is no winner between early fusion and late fusion with respect to the F-average, AP and accuracy. Due to the fact that sentiment is much more abstract and extremely challenging to be represented from visual data, elaborative designs for feature selection and multi-feature fusion strategies are needed.

Another interesting observation is that the results of different approaches on the Sen-

Table 5.11: Performance of different visual features for sentiment analysis on Sentibank dataset. The best result in each kind of visual features is underlined, and the best one of all the approaches is marked in bold.

Feature		F-positive	F-negative	F-average	AP	Accuracy
Low-Level	Color Histogram	0.845	0.069	0.457	0.803	0.735
	GIST	0.834	0.028	0.431	0.752	0.738
	LBP	0.875	0.000	0.437	0.788	0.735
	BoVW	0.867	0.078	0.473	0.807	0.742
Middle-Level	Classemes	0.853	0.175	0.514	0.809	0.751
	Attribute	0.853	0.095	0.474	0.806	0.748
	SentiANP	0.866	0.125	0.495	**0.823**	0.768
Aesthetic	AF	0.868	0.000	0.434	0.811	0.768
	PoA	**0.878**	0.084	0.481	0.806	0.764
Fusion	LV-Early	0.857	0.050	0.453	0.790	0.751
	LV-Late	0.873	0.000	0.436	0.799	0.774
	MV-Early	0.861	**0.182**	**0.521**	0.794	0.761
	MV-Late	0.871	0.081	0.476	0.812	0.775
	AV-Early	0.867	0.054	0.461	0.787	0.768
	AV-Late	0.873	0.000	0.436	0.814	0.774
	V-Early	0.871	0.128	0.499	0.795	0.776
	V-Late	0.875	0.000	0.437	0.813	**0.778**

tibank dataset are not stable. This is because the dataset is extremely biased on positive instances in both training and testing sets. Since the positive instances is much more than the negative ones, the learned model will focus on the accuracy of positive instances. This results in a very low F-negative for all the methods in Table 5.11. Another reason may be that the size of the Sentibank dataset is small. The performance of model learning and

Table 5.12: Prediction results on the MVSA image dataset using models learned from different datasets with various visual features. For each kind of feature, the best results are marked in bold.

Feature	Training Set	F-positive	F-negative	F-average	AP	Accuracy
BoVW	ImgTweet	0.737	**0.357**	0.547	0.687	0.627
	Sentibank	**0.788**	0.112	0.450	0.667	0.657
	MVSA	0.775	0.354	**0.565**	**0.740**	**0.667**
SentiANP	ImgTweet	0.752	0.336	0.544	0.742	0.639
	Sentibank	0.786	0.107	0.447	0.713	0.655
	MVSA	**0.790**	**0.378**	**0.584**	**0.779**	**0.687**
AF	ImgTweet	0.749	**0.390**	**0.570**	**0.758**	0.644
	Sentibank	**0.792**	0.016	0.404	0.669	0.656
	MVSA	0.785	0.181	0.483	0.755	**0.659**

testing is sensitive to the settings of the experiments. Thus a large and balanced dataset is needed to build and fairly evaluate a sentiment analysis system.

5.3.2 Cross Dataset Performance

Similar to the text sentiment analysis, domain shift is also an important problem for the generalization capacity of learned models in image sentiment analysis. In this section, we will adopt the models learned from ImgTweet and Sentibank on our constructed dataset. Due to the limitation of space, we will only test the most robust low-level, middle-level and aesthetic features, respectively BoVW, SentiANP and AF. The results are showed in Table 5.12. For the BoVW and SentiANP features, we can see that the performance degrades a lot using classifiers from a dataset different from the testing data due to the problem of domain shift. For aesthetic feature, the model learned from the ImgTweet

Table 5.13: Performace of multi-view sentiment analysis on Sentibank and MVSA datasets using two fusion strategies. The best results are marked in bold.

Dataset	Method	F-positive	F-negative	F-average	AP	Accuracy
Sentibank	T-V-Early	0.858	0.234	**0.546**	**0.865**	0.761
	T-V-Late	**0.873**	0.107	0.490	0.864	**0.778**
	TF	0.835	**0.251**	0.543	0.849	0.738
	SentiANP	0.866	0.125	0.495	0.823	0.768
MVSA	T-V-Early	0.805	**0.582**	**0.693**	**0.889**	0.734
	T-V-Late	**0.821**	0.543	0.682	0.887	**0.743**
	TF	0.792	0.547	0.670	0.870	0.715
	SentiANP	0.790	0.378	0.584	0.779	0.687

dataset performs comparably well on our constructed MVSA dataset. The reasons are two-fold. Firstly, aesthetic feature represents the more abstract aspect of visual content, and thus the domain gap is smaller than other features. The accuracy of the classifier learned from the Sentibank dataset can also approach the one of the MVSA model. Secondly, ImgTweet contains more and balanced training instances, which is significantly important for learning a model with good generalization capability.

5.4 Results on Multi-view Data

The experiments in this section are conducted on the two available multi-view datasets, Sentibank and our MVSA.

5.4.1 Performance of Fusion Strategies

In this section, we examine the effectiveness of simple early and late fusion approaches on multi-view data. T-V-Early and T-V-Late respectively represent the early fusion and late fusion of textual and visual features. The experimental results in section 5.3.1 have showed that fusing more features may hurt the performance in sentiment analysis. Thus we only fuse the TF and SentiANP features which are two representative textual and visual features in the literature. The results on two multi-view datasets are showed in Table 5.13, where the performance of individual features is also included. We can see that jointly utilizing textual and visual information in the tweets by linearly fusing the features can boost the performance significantly. Comparing Table 5.13 with Table 5.9 and Table 5.11, T-V-Late outperforms all the approaches in single view sentiment analysis, even the results using multiple visual features.

5.4.2 Performance of Joint Feature Learning

In this section, we will evaluate the learned features, which are the outputs of different layers in the M-DBM. The results on two datasets are showed in Table 5.14, where h^{1t} and h^{2t} are the first and second layers in the textual pathway, h^{1v} and h^{2v} denote the first and second layers in the visual pathway, and h^{J} is the final joint layer. We also list the performance of TF which is the input of the textual pathway in M-DBM. For the reason of consistency, TF in M-DBM adopts the vocabulary generated from the STS dataset. Thus, the TF on Sentibank dataset in Table 5.14 is different from the one in Table 5.13, which leverages a vocabulary from the Sentibank dataset.

We can see that the second layer performs better than the first layer in both textual and visual pathways. Finally, the joint layer achieves the best performance on both Sentibank and MVSA datasets. Jointly considering the information from the two pathways, h^{J} significantly improves the performance on two datasets over the TF feature. This is consistent

Table 5.14: Performace of multi-view sentiment analysis on Sentibank and MVSA datasets using feastures of different layers in the learned M-DBM model. The best results are marked in bold.

Dataset	Layer	F-positive	F-negative	F-average	AP	Accuracy
Sentibank	h^{1t}	0.864	0.101	0.483	0.760	0.761
	h^{2t}	**0.873**	0.000	0.436	0.771	0.764
	h^{1v}	0.762	**0.189**	0.476	0.772	0.632
	h^{2v}	0.792	0.180	0.486	0.777	0.668
	h^J	0.858	0.160	**0.509**	**0.784**	**0.768**
	TF	0.861	0.026	0.444	0.761	0.757
MVSA	h^{1t}	0.774	0.358	0.566	0.773	0.666
	h^{2t}	0.804	0.291	0.548	0.858	0.693
	h^{1v}	0.733	0.388	0.560	0.729	0.628
	h^{2v}	0.727	0.370	0.548	0.732	0.629
	h^J	**0.825**	**0.535**	**0.680**	**0.865**	**0.746**
	TF	0.792	0.547	0.670	0.870	0.715

with the observation in [45]. In addition, the inputs of the M-DBM model are several low-level features. Comparing to the results of low-level features in Table 5.9 and Table 5.11, we can see that the performance is also boosted significantly with the learned joint feature. This indicates that the correlation between the two views can be captured and represented by the M-DBM. This is helpful in predicting complicated human affection. Compared to the results of T-V-Early and T-V-Late in Table 5.13, the performance of h^J is slightly worse than the fusion strategies which also adopt information of two views. This may be caused by the different properties between Twitter messages and the data for learning the M-DBM which is collected from Flickr. In other words, the learned M-DBM reflects the correlations between text and visual on Flickr images, which may be different from Twitter

Table 5.15: Prediction results on MVSA dataset using models learned from the Sentibank and MVSA datasets with h^J feature. The best results are marked in bold.

Training set	F-positive	F-negative	F-average	AP	Accuracy
Sentibank (h^J)	0.793	0.000	0.396	0.698	0.657
MVSA (h^J)	**0.825**	**0.535**	**0.680**	**0.865**	**0.746**

messages. Thus the joint feature extracted from the M-DBM may not perform best. Even though, h^J outperforms the SentiANP feature which is dedicatedly designed for human affective computing. Another advantage is that the M-DBM feature (2,048 dimensions) is more compact than T-V-Early and T-V-Late using a 4,000 dimensional feature. In general, Table 5.14 shows encouraging performances on multi-view sentiment analysis, which is worthy of further investigation.

Finally, we show the results on the MVSA testing set using classifiers learned from the Sentibank training set and the MVSA training set respectively in Table 5.15. We can see that the model learned using the Sentibank data is much worse than the one learned from the MVSA. Besides the shift of domain from the training set to the testing set, insufficient and unbalanced training data in the Sentibank dataset affect the performance of the learned model. Thus, a more complete and large-scale dataset is needed for developing and evaluating multi-view sentiment analysis systems.

Chapter 6

Conclusion and future work

In this chapter, we summarize the major contributions and achievements of this thesis. We will also list some interesting future directions.

6.1 Summary of Contributions

In this thesis, we have contributed to tweet sentiment analysis in three aspects.

- In Chapter 3, we have provided a general pipeline for sentiment analysis on single-view or multi-view social data. We have also provided a systematic survey on different approaches for sentiment analysis including different feature representations of various media types.

- In Chapter 4, we have introduced a new dataset called MVSA consisting of multi-view tweets for sentiment analysis. To the best of our knowledge, it has been the largest dataset dedicatedly constructed for multi-view sentiment analysis. Annotations for both texts and images are provided.

- In Chapter 5, the state-of-the-art approaches are extensively evaluated and compared through a comprehensive set of experiments performed on public available datasets and our constructed dataset. Several important issues such as vocabulary in BoW and domain shift which may affect the performance of adopted approaches are studied. The results show that a large and balanced dataset is essential for model learning and evaluation. In addition, the correlations embedded in multiple views in Twitter messages have been demonstrated to be helpful in understanding the human affection. This provides a promising way for further investigating multi-view data.

6.2 Future work

Besides the sentiment analysis discussed in this thesis, there are still several interesting and important issues that can be further investigated with the help of our contributed dataset. In addition, our dataset can be enriched to cover more fine-grained labels. We list several future works here.

- We have showed that there are many inconsistent labels between the text view and image view in the collected tweets. This suggests that future research should pay particular attention on the differentiation of emotional context with other contexts between two views, so that we can appropriately leverage the information from two views.

- The joint feature exploring the correlations between two views has been demonstrated to be helpful. However, the advantage seems to be not so obvious comparing to the traditional fusion strategies. This raises the issue that how to keep the specific property of each view while modeling their contexts. It is important to purse an intermediate status, so that the discriminative information of single view will be kept as much as possible, meanwhile, the extracted contexts are not harmful.

- Our dataset is constructed by setting a global sentiment for each tweet without considering the entity-level sentiment. In addition, mixed sentiments which may appear in certain tweets are ignored in current work. Since the annotation on entity-level is so expensive, an appropriate interactive annotation tool is needed. For example, we can pre-filter the non-informative messages to reduce the number of messages.

References

[1] Subhabrata Bhattacharya, Behnaz Nojavanasghari, Tao Chen, Dong Liu, Shih-Fu Chang, and Mubarak Shah. Towards a comprehensive computational model for aesthetic assessment of videos. In *ACM MM*, 2013.

[2] John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification. In *In ACL*, 2007.

[3] Damian Borth, Rongrong Ji, Tao Chen, Thomas Breuel, and Shih-Fu Chang. Large-scale visual sentiment ontology and detectors using adjective noun pairs. In *ACM MM*, 2013.

[4] Luigi Di Caro and Matteo Grella. Sentiment analysis via dependency parsing. *Computer Standards and Interfaces*, 35(5):442–453, 2013.

[5] Yan-Ying Chen, Tao Chen, Winston H. Hsu, Hong-Yuan Mark Liao, and Shih-Fu Chang. Predicting viewer affective comments based on image content in social media. In *ICMR*, 2014.

[6] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[7] Kushal Dave, Steve Lawrence, and David M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *WWW*, 2003.

[8] Nicholas A. Diakopoulos and David A. Shamma. Characterizing debate performance via aggregated twitter sentiment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2010.

[9] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[10] Anindya Ghose, Panagiotis Ipeirotis, and Arun Sundararajan. Opinion mining using econometrics: A case study on reputation systems. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 2007.

[11] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. *Processing*, pages 1–6, 2009.

[12] A. Hanjalic. Extracting moods from pictures and sounds: Towards truly personalized. *IEEE Signal Processing Magazine*, 23(2):90–100, 2006.

[13] Vasileios Hatzivassiloglou and Kathleen McKeown. Predicting the semantic orientation of adjectives. In *ACL*, 1997.

[14] Marko Heikkilä and Matti Pietikäinen. A texture-based method for modeling the background and detecting moving objects. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(4):657–662, 2006.

[15] Geoffrey Hinton and Ruslan Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313:504 – 507, 2006.

[16] Georey Hinton. A practical guide to training restricted boltzmann machines. Technical report, 2010.

[17] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.

[18] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *SIGKDD*, 2004.

[19] Xia Hu, Lei Tang, Jiliang Tang, and Huan Liu. Exploiting social relations for sentiment analysis in microblogging. In *WSDM*, 2013.

[20] Johannes Itten. *The art of color: the subjective experience and objective rationale of color*. Wiley, 1974.

[21] Yu-Gang Jiang, Chong-Wah Ngo, and Jun Yang. Towards optimal bag-of-features for object categorization and semantic video retrieval. In *CIVR*, 2007.

[22] Yu-Gang Jiang, Baohan Xu, and Xiangyang Xue. Predicting emotions in user-generated videos. In *AAAI*, 2014.

[23] Xiao-Yuan Jing, Ruimin Hu, Yang-Ping Zhu, Shan-Shan Wu, Chao Liang, and Jing-Yu Yang. Intra-view and inter-view supervised correlation analysis for multi-view feature learning. In *AAAI*, 2014.

[24] Dhiraj Joshi, Ritendra Datta, Elena A. Fedorovskaya, Quang-Tuan Luong, James Ze Wang, Jia Li, and Jiebo Luo. Aesthetics and emotions in images. *IEEE Signal Process. Mag.*, 28(5):94–115, 2011.

[25] Michael Kass and Justin Solomon. Smoothed local histogram filters. *ACM Trans. Graph.*, 29(4):100:1–100:10, 2010.

[26] Li-Jia Li, Hao Su, Eric P. Xing, and Fei-Fei Li. Object bank: A high-level image representation for scene classification and semantic feature sparsification. In *NIPS*, 2010.

[27] Ying Li, Zijian Zheng, and Honghua (Kathy) Dai. Kdd cup-2005 report: Facing a great challenge. *SIGKDD Explor. Newsl.*, 7(2):91–99, 2005.

[28] Kar Wai Lim and Wray Buntine. Twitter opinion topic model: Extracting product opinions from tweets by leveraging hashtags and sentiment lexicon. In *CIKM*, 2014.

[29] Bing Liu. *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2012.

[30] David G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, 1999.

[31] Gareth Loy and Jan-Olof Eklundh. Detecting symmetry and symmetric constellations of features. In *ECCV*, 2006.

[32] Gareth Loy and Alexander Zelinsky. Fast radial symmetry for detecting points of interest. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(8):959–973, 2003.

[33] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *ACL*, 2011.

[34] Jana Machajdik and Allan Hanbury. Affective image classification using features inspired by psychology and art theory. In *ACM MM*, 2010.

[35] Hye-Jin Min and Jong C. Park. Identifying helpful reviews based on customer's mentions about experiences. *Expert Syst. Appl.*, 39(15):11830–11838, 2012.

[36] A. Moreo, M. Romero, J.L. Castro, and J.M. Zurita. Lexicon-based comments-oriented news sentiment analyzer system. *Expert Syst. Appl.*, 39(10):9166–9180, 2012.

[37] Tony Mullen and Nigel Collier. Sentiment analysis using support vector machines with diverse information sources. In *EMNLP*, 2004.

[38] Naila Murray, Luca Marchesotti, and Florent Perronnin. AVA: A large-scale database for aesthetic visual analysis. In *CVPR*, 2012.

[39] Eric Nowak, Frédéric Jurie, and Bill Triggs. Sampling strategies for bag-of-features image classification. In *ECCV*, 2006.

[40] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

[41] Bo Pang and Lillian Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, 2004.

[42] Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, 2005.

[43] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, 2007.

[44] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *EMNLP*, 2002.

[45] Lei Pang and Chong-Wah Ngo. Multimodal learning with deep boltzmann machine for emotion prediction in user generated videos. In *ICMR*, 2015.

[46] Nataliia Plotnikova, Micha Kohl, Kevin Volkert, Andreas Lerner, Natalie Dykes, Heiko Ermer, and Stefan Evert. KLUEless: Polarity classification and association. *SemEval 2015 workshop*, 2015.

[47] Jonathon Read and John Carroll. Weakly supervised techniques for domain-independent sentiment classification. In *CIKM Workshop on TSA'09*, 2009.

[48] Sara Rosenthal, Preslav Nakov, Svetlana Kiritchenko, Saif M Mohammad, Alan Ritter, and Veselin Stoyanov. SemEval-2015 Task 10: sentiment analysis in twitter. *SemEval 2015 workshop*, 2015.

[49] Hassan Saif, Miriam Fernandez, Yulan He, and Harith Alani. Evaluation datasets for twitter sentiment analysis: A survey and a new dataset, the sts-gold. *ESSEM workshop*, 2013.

[50] Ruslan Salakhutdinov and Geoffrey Hinton. Deep Boltzmann machines. In *AI Statistics*, 2009.

[51] Ruslan Salakhutdinov and Geoffrey Hinton. Replicated softmax: an undirected topic model. In *NIPS*, 2010.

[52] Yohei Seki, David Kirk Evans, Lun-Wei Ku, Hsin-Hsi Chen, Noriko Kando, and Chin-Yew Lin. Overview of opinion analysis pilot task at NTCIR-6. In *Proceedings of the Workshop Meeting of the National Institute of Informatics (NII) Test Collection for Information Retrieval Systems (NTCIR)*, 2007.

[53] David A. Shamma, Lyndon Kennedy, and Elizabeth F. Churchill. Tweet the debates: Understanding community annotation of uncollected sources. In *Proceedings of the First SIGMM Workshop on Social Media*, 2009.

[54] L.G. Shapiro and G.C. Stockman. *Computer Vision.* Prentice Hall, 2001.

[55] Eli Shechtman and Michal Irani. Matching local self-similarities across images and videos. In *CVPR*, 2007.

[56] Stefan Siersdorfer, Enrico Minack, Fan Deng, and Jonathon Hare. Analyzing and predicting sentiment of images on the social web. In *ACM MM*, 2010.

[57] Arnold W. M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(12):1349–1380, 2000.

[58] Benjamin Snyder and Regina Barzilay. Multiple aspect ranking using the good grief algorithm. In *In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL*, 2007.

[59] Michael Speriosu, Nikita Sudan, Sid Upadhyay, and Jason Baldridge. Twitter polarity classification with label propagation over lexical links and the follower graph. In *EMNLP Workshop*, 2011.

[60] Nitish Srivastava and Ruslan Salakhutdinov. Multimodal learning with deep boltzmann machines. *Journal of Machine Learning Research*, 15(1):2949–2980, 2014.

[61] Moritz Sudhof, Andrés Goméz Emilsson, Andrew L. Maas, and Christopher Potts. Sentiment expression conditioned by affective transitions and social forces. In *SIGKDD*, 2014.

[62] Xiaoshuai Sun, Hongxun Yao, Rongrong Ji, and Shaohui Liu. Photo assessment based on computational visual attention model. In *ACM Multimedia*, 2009.

[63] Jussi Tarvainen, Mats Sjöberg, Stina Westman, Jorma Laaksonen, and Pirkko Oittinen. Content-based prediction of movie style, aesthetics, and affect: Data set and baseline experiments. *IEEE Transactions on Multimedia*, 16(8):2085–2098, 2014.

[64] Mike Thelwall, Kevan Buckley, and Georgios Paltoglou. Sentiment strength detection for the social web. *J. Am. Soc. Inf. Sci. Technol.*, 63(1):163–173, 2012.

[65] Matt Thomas, Bo Pang, and Lillian Lee. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In *Proceedings of EMNLP*, pages 327–335, 2006.

[66] Lorenzo Torresani, Martin Szummer, and Andrew Fitzgibbon. Efficient object category recognition using classemes. In *ECCV*, 2010.

[67] Peter D. Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *ACL*, 2002.

[68] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas S. Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *CVPR*, 2010.

[69] Y. Wang, Q. Dai, R. Feng, and Y.-G. Jiang. Beauty is here: Evaluating aesthetics in videos using multimodal features and free training data. In *ACM MM*, 2013.

[70] Casey Whitelaw, Navendu Garg, and Shlomo Argamon. Using appraisal groups for sentiment analysis. In *CIKM*, 2005.

[71] Janyce Wiebe and Claire Cardie. Annotating expressions of opinions and emotions in language. language resources and evaluation. In *Language Resources and Evaluation (formerly Computers and the Humanities*, 2005.

[72] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, 2005.

[73] Wenxuan Xie, Yuxin Peng, and Jianguo Xiao. Cross-view feature learning for scalable social image analysis. In *AAAI*, 2014.

[74] Can Xu, Suleyman Cetintas, Kuang-Chih Lee, and Li-Jia Li. Visual sentiment prediction with deep convolutional neural networks. *CoRR*, 2014.

[75] Li Xu, Qiong Yan, Yang Xia, and Jiaya Jia. Structure extraction from texture via relative total variation. *ACM Trans. Graph.*, 31(6):139:1–139:10, 2012.

[76] Tao Xu, Qinke Peng, and Yinzhao Cheng. Identifying the semantic orientation of terms using S-HAL for sentiment analysis. *Knowl.-Based Syst.*, 35:279–289, 2012.

[77] Yun Yang, Peng Cui, Wenwu Zhu, H. Vicky Zhao, Yuanyuan Shi, and Shiqiang Yang. Emotionally representative image discovery for social events. In *ICMR*, 2014.

[78] Quanzeng You and Jiebo Luo. Towards social imagematics: Sentiment analysis in social multimedia. In *MDMKDD*, 2013.

[79] Quanzeng You, Jiebo Luo, Hailin Jin, and Jianchao Yang. Robust image sentiment analysis using progressively trained and domain transferred deep networks. In *AAAI*, 2015.

[80] Felix Yu, Liangliang Cao, Rogerio Feris, John Smith, and Shih-Fu Chang. Designing category-level attributes for discriminative visual recognition. In *CVPR*, 2013.

[81] Jianbo Yuan, Sean Mcdonough, Quanzeng You, and Jiebo Luo. Sentribute: Image sentiment analysis from a mid-level perspective. In *WISDOM '13*, 2013.

[82] Sicheng Zhao, Yue Gao, Xiaolei Jiang, Hongxun Yao, Tat-Seng Chua, and Xiaoshuai Sun. Exploring principles-of-art features for image emotion recognition. In *ACM MM*, 2014.

[83] Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. Comparing twitter and traditional media using topic models. In *ECIR*, 2011.