Multimedia Content Understanding by Analyzing Perceived Signals: Face, Sentiment and Emotion
基於人臉、情感和情緒等感知信號的多媒體內容解析

Submitted to
Department of Computer Science
電腦科學系
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
哲學博士學位

by

Pang Lei
龐磊

November 2015
二零一五年十一月
ABSTRACT

Content understanding of unconstrained user-generated videos is a challenging task, due to the difficulty in mining semantics out of rich visual-audio signals in addition to the sparse and subjective text descriptions. This thesis addresses the challenge by multi-modal analysis of three perceived signals: face, sentiment and emotion, for cross-media exploratory browsing. Specifically, we study approaches in naming faces for video browsing, predicting sentiment underlying short video clips for question-answering, and emotion categorization for cross-modal retrieval. The results of analysis for the three signals are then integrated into a system for search navigation and exploration in large video archive.

Face naming is challenging due to large variations of face appearances in unconstrained videos. Instead of relying on accurate face labels for supervised learning, a rich set of relationships automatically derived from video content and knowledge from image domain and social cues is leveraged for unsupervised face labeling. The relationships refer to the appearances of faces under different spatio–temporal contexts and their visual similarities. The knowledge includes Web images weakly tagged with celebrity names and the celebrity social networks. The relationships and knowledge are elegantly encoded using conditional random field (CRF) for label inference. Two versions of face annotation are considered: within-video and between-video face labeling. The former addresses the problem of incomplete and noisy labels in metadata, where null assignment of names is allowed – a problem seldom been considered in the literature. The latter further rectifies the errors in metadata, specifically to correct false labels and annotate faces with missing names in the metadata of a video, by considering a group of socially connected videos for joint label inference.
Popular web videos, especially videos about hotly-debated news events, often express sentiment. We exploit multi-modal sentiment analysis of videos for question-answering (QA), given that there are large portions of opinion-oriented questions in QA archives. Some of these questions are better answered by videos than texts, due to the vivid display of emotional signals visible through face expression and speaking tone. The difficulty of the multimedia opinion question answering lies in two aspects. On one hand, a potential answer of duration 60 seconds may be embedded in a video of 10 minutes, resulting in degraded user experience compared to reading answers in text only. On the other hand, a text-based opinion question may be short and vague while the video answers could be verbal, less structured in grammar and noisy because of errors in speech transcription. Direct matching of words or syntactic analysis of sentence structure, such as adopted by factoid and how-to question-answering, is unlikely to find the video answers. The first problem, specifically the answer localization, is addressed by audio-visual analysis of the emotional signals in videos for locating video segments likely expressing opinions. The second problem, questions and answers matching, is tackled by a deep architecture that non-linearly matches text words in questions and speeches in videos.

For emotion prediction, we explore the learning of highly non-linear relationships that exist among low-level features across different modalities. Using Deep Boltzmann Machine (DBM), a joint density model over the modalities, is developed. The model is trained directly using user generated videos without any labeling efforts. While the model learns a joint representation over multimodal inputs, training samples in absence of certain modalities can also be leveraged. More importantly, the joint representation enables emotion-oriented cross-modal retrieval, for example, retrieval of videos using text query “crazy cat”. The model does not restrict the types of input and output, and hence in principle, emotion
prediction and retrieval on any combinations of media are feasible.

Finally, the three pieces of works are integrated into a system for exploratory-based opinion QA. The system creates hyperlinking connecting video clips sharing similar perceived signals. The system is designed in such a way that users can easily navigate the search results, exploring along the hyperlinks to different aspects of information based on either topic, person or emotion. The system is experimented on BBC videos of more than 3,000 hours in duration.
CITY UNIVERSITY OF HONG KONG
Qualifying Panel and Examination Panel

Surname: PANG
First Name: Lei
Degree: Doctor of Philosophy
College/Department: Department of Computer Science

The Qualifying Panel of the above student is composed of:

Supervisor(s)
Prof. NGO Chong Wah
Department of Computer Science
City University of Hong Kong

Qualifying Panel Member(s)
Dr. CHAN Antoni Bert
Department of Computer Science
City University of Hong Kong

Dr. WONG Hau San
Department of Computer Science
City University of Hong Kong

This thesis has been examined and approved by the following examiners:

Prof. NGO Chong Wah
Department of Computer Science
City University of Hong Kong

Prof. ZHU Jian Hua Jonathan
Department of Media and Communication
City University of Hong Kong

Dr. HUET Benoit
Multimedia Information Processing Group
Eurecom

Prof. SATOH Shin'ichi
Digital Content and Media Sciences Research Division
National Institute of Informatics
ACKNOWLEDGEMENTS

Working as a Ph.D. student in VIREO was a magnificent as well as challenging experience to me. In all these years, many people were instrumental directly or indirectly in shaping up my academic career. It was hardly possible for me to thrive in my doctoral work without the precious support of these personalities. Here is a small tribute to all those people.

First of all, I wish to thank my supervisor Prof. Chong-Wah Ngo for introducing me to the world of multimedia search and computing. It was only due to his valuable guidance, cheerful enthusiasm and ever-friendly nature that I was able to complete my research work in a respectable manner. I am also greatly appreciate the extra one year that I was given to finish my Ph.D. study. The experience of working with such a true expert values much beyond this Ph.D.

I would like to thank my examiners, Prof. Shin‘ichi Satoh, Prof. Jonathan Zhu and Dr. Benoit Huet, for their precious time and valuable feedbacks. I am also grateful to my qualifying panel members, Dr. Hau San Wong and Dr. Antoni Bert Chan, for their constructive suggestions and comments on my research reports.

I am privileged for having Xiao Wu, Xiao-Yong Wei, Yu-Gang Jiang, Hung-Khoon Tan, Wan-Lei Zhao, Shi-Ai Zhu, Song Tan, Ting Yao, Wei Zhang, Chun-Chet Tan, Zhi-Neng Chen, Hao Zhang, Yi-Jie Lu and Jing-Jing Chen as my colleagues, who have provided great company and inspiration during all these years in Hong Kong. Especially, I would like to thank Shi-Ai Zhu for his generous help and steadfast support, which is more than precious. Wei Zhang, Song Tan, Hung-Khoon Tan, Ting Yao and Yu-Gang Jiang also deserve my special thanks for their efforts and collaborations on my research papers.

Finally, and most importantly, I would like to thank my wife Ya-Qun Dong. Her support, encouragement, quiet patience and unwavering love were undeniably
the bedrock upon which the past nine years of my life have been built. She, as a
gift from God, made my life colorful and happy. I greatly appreciate my parents
for their faith in me and allowing me to be as ambitious as I wanted. Deepest
thanks to everything they did and every sacrifice they made on my behalf. I thank
my sister, sister-in-law and nephew for taking care of parents when I am pursing
the degree in Hong Kong. I thank my pet dogs, Xiaoqi and Ponyo, for happy
company in the last two years.
# TABLE OF CONTENTS

Title Page ii

Abstract ii

Acknowledgements v

Table of Contents vii

List of Figures xi

List of Tables xv

## 1 Introduction

1.1 Understanding Perceived Signals 1
   1.1.1 Celebrity Naming 1
   1.1.2 Sentiment Analysis for Question Answering 2
   1.1.3 Emotion Prediction 3
   1.1.4 Exploratory Search 4
1.2 Overview of Approaches and Contributions 5
1.3 Organization 8
1.4 Publication 8

## 2 Literature Review

2.1 Face Naming 9
   2.1.1 Model-based Approaches 9
   2.1.2 Search-based Approaches 10
   2.1.3 Constrained Clustering-based Approaches 11
2.2 Multimedia Question Answering 12
   2.2.1 Factoid Question Answering 12
   2.2.2 How-to Question Answering 13
   2.2.3 Multimodal Question Answering 13
   2.2.4 Lexical and Stylistic Gap in QA 14
2.3 Affective Computation 15
   2.3.1 Affection in Textual Documents 15
   2.3.2 Affection in Images 16
2.3.3 Affection in Music
2.3.4 Affection in Video
2.3.5 Comparisons

3 Unsupervised Celebrity Face Naming in Web Videos
3.1 Relationship Modeling
  3.1.1 Problem Definition and Notation
  3.1.2 Unary Potential
  3.1.3 Pairwise Potential
    3.1.3.1 Spatial Relationship
    3.1.3.2 Temporal Relationship
    3.1.3.3 Visual Relationship
  3.1.4 Complexity Analysis
3.2 Leveraging Social Cues
  3.2.1 Social-driven Graph Splitting
  3.2.2 Name-to-name Social Relation
  3.2.3 Algorithm Implementation
3.3 Experiments
  3.3.1 Dataset and Evaluation Metrics
    3.3.1.1 Dataset
    3.3.1.2 Supporting Image Dataset and Features
    3.3.1.3 Performance Evaluation
  3.3.2 Parameter Sensitivity
  3.3.3 Within-video Face Naming
  3.3.4 Between-video Face Naming
  3.3.5 Performance Comparison
  3.3.6 Discussion
3.4 Summary

4 Opinion Question Answering with Sentimental Clips
4.1 Sentiment Detection
  4.1.1 Extracting Talking Tracks
  4.1.2 Sentiment Classification
4.2 Locate Opinion Clip
  4.2.1 Features
  4.2.2 EM algorithm
4.3 Question-Answering by Deep Learning
5 Deep Multimodal Learning for Affective Analysis and Retrieval

5.1 Deep Network Design
5.2 Multimodal Joint Representation
  5.2.1 Visual Pathway
  5.2.2 Auditory Pathway
  5.2.3 Textual Pathway
  5.2.4 Joint Representation
5.3 Networking Learning and Inferencing
  5.3.1 Approximate Network Learning
  5.3.2 Joint Representation Inferring
  5.3.3 Discussions
5.4 Experiment: Affective Analysis
  5.4.1 Model Learning
  5.4.2 Video Emotion Detection
  5.4.3 Sentiment Analysis on Twitter Messages
5.5 Experiment: Retrieval
  5.5.1 Video Retrieval
  5.5.2 Cross-Modal Retrieval
5.6 Summary

6 Constellation Browser for Exploratory Search

6.1 Technology
  6.1.1 Offline Preprocessing
    6.1.1.1 Video Processing for Link Construction
    6.1.1.2 Deep Match Learning for Opinion QA
6.1.2 Online Constellation Searching

6.2 User Study
   6.2.1 Dataset
   6.2.2 Evaluation and Result

6.3 Summary

7 Conclusion and Future Directions
   7.1 Summary of Contributions
   7.2 Future Directions

References

List of Publications
LIST OF FIGURES

3.1 An example of Web video illustrating the challenge of associating the names (italic) in metadata with the detected faces (with bounding boxes) in the video. Among the 14 faces of 4 celebrities (Hillary Clinton, Barack Obama, Wolf Blitzer and John Edwards), only 4 faces (dotted bounding boxes) of 2 celebrities (Hillary Clinton and Barack Obama) are mentioned in the metadata. In addition, only two (Hillary Clinton and Barack Obama) out of three celebrities who are mentioned appear in the video.

3.2 Within-video naming constructs a graph modeling the face-to-name and face-to-face relationships among the faces and names found in a video ($V_1$). By social network, between-video labeling expands the graph by connecting to the graphs of two other videos ($V_2$ and $V_3$) that share social relations. The expanded graph is additionally modeled with name-to-name relationship inferred from the social network.

3.3 An example of graph depicting the modeling of relationships for face naming as an optimization problem. The objective function is to maximize the probability of assigning the right names or labels (denoted as $y_i$) to faces based upon the unary and pairwise potentials defined by various relationships.

3.4 Between-video naming: the process of constructing social networks and performing CRF on multiple smaller graphs.

3.5 Sensitivity of the parameters $\eta$ (left) and $\delta$ (right) towards the accuracy and precision of face naming.

3.6 Performance comparison of eight different approaches on Cele-WebV+ dataset.

3.7 Time cost (seconds) of various methods on Cele-WebV.

3.8 Samples of “noisy” (bottom) versus “clean” (top) faces, due to different effects: pose, illumination, resolution, make-up, occlusion, aging, drawing (from left to right).
3.9 True and false positives by CRF-G: (a) the face $f_3$ is strongly connected to the label “Barack Obama”, and indirectly influences the labeling of three other faces suffering from changes in face appearances; (b) two visually similar faces ($f_1$: Alexandra Burke, $f_2$: Beyonce Knowles) result in false labeling of $f_2$.

3.10 The tradeoff between t-precision and t-recall when thresholding the results of CRF-G on Cele-WebV*.

4.1 An example of opinion question and video answer. In this chapter, we are interested in locating a segment in the video (the highlighted box) that has an answer to the question. The challenges include locating a clip where an opinion holder expresses views of the question from a lengthy video, and the fact that there are very few overlapping words between text question and speech transcripts (underlined) for reliable matching.

4.2 Framework of the proposed system. The boxes in green color indicate potential video segments to be selected for question answering.

4.3 An example of DAG for the topic “Obamacare”. The label on each subtopic is manually defined for illustration purpose. The symbols “Q:” and “A:” indicate the words from question and answer domains respectively.

4.4 Illusion of the neural network based on hierarchical topic structure.

4.5 Performance comparison of five approaches on opinion question-answering, measured in terms of average nDCG@10.

4.6 Trends of ranking-based loss of DeepHPam and DeepMatch on the topics “Obamacare” and “Snowden”. The loss is summed over the entire training set. The learning rates is 0.01 and the coefficient of L2 regularization term is 0.0001. The training losses of other topics follow the similar trends.

4.7 Example of opinion clips retrieved by DeepHPam. The keyframes and partial transcripts of the opinion clips are shown for illustration purpose.
5.1 Examples of emotional images and videos with their associated
tags and titles from Twitter and YouTube. The left image in the
first row describes the destruction caused by Hurricane Sandy and
the right one describes the Manta roller coaster in the seaworld
Orlando. The textual descriptions for these two images are almost
the same and the implicit emotions can only be predicted by the
visual information. In contrast, emotion classification in the lower
two videos with similar visual appearances is performed using the
strong clues in the textual captions and auditory information.

5.2 The multimodal DBM that models the joint distribution over vi-
sual, auditory and textual features. All layers but the first (bottom)
layers use standard binary units. Gaussian RBM model is used to
model the distributions over the visual and auditory features. Repli-
cated Softmax topic model is applied on the textual features.

5.3 Confusion matrix based on (a) joint representation and (b) fusion
results on the VideoEmotion dataset.

5.4 Prediction accuracies. SentiBank is the attribute feature proposed
in [1]. E-MDBM-V and E-MDBM-A represent the joint represen-
tation generated by our proposed E-MDBM using only the visual
or auditory signals as inputs. Similarly, E-MDBM-VA indicates the
joint representation using both visual and auditory modalities.

5.5 Prediction accuracies on the ImageTweets. E-MDBM-V and E-
MDBM-T are the classifiers trained on the joint representation gen-
erated by using only the visual or textual information through E-
MDBM. E-MDBM-VT is trained on the joint representations over
both visual and textual modalities. MDBM represents the joint
representations over both visual and textual modalities but based
on the architecture proposed in [2]. Meanwhile, SentiBank repre-
sents the classifiers trained on the scores of the concepts classifirs
in [1]. Lexicon represents the Naive Bayes classifiers trained based
on SentiStrength [3]. AF represents the classifiers trained on the
art features proposed in [4].
5.6 Examples of video queries. The caption for the first example (top-left) describes disable kitten, but the video expresses a “joy” emotion with audio-visual effect. The two videos in the second row show counter examples. The video on the left has a spider but is not mentioned by caption, as opposed to the video on the right where spider is mentioned in caption but dose not appear in video. The query (top-right) expresses consistent emotion and semantics across visual, auditory and textual modalities.

6.1 Framework of Constellation Browser. The online searching is conducted based on the “Deep Match” model. The celebrity, topic and emotion links in constellation presentation are constructed based on the celebrity inverted index, topic and emotion link lookup tables respectively. After exploring the searching result, users can refine queries and search again.

6.2 Example Constellation on the “Cola Boy” in Sichuan Earthquake. The surrounding rectangles of nodes represent the node types: green for positive sentimental clips, red for negative sentimental clips and black for factoid clips. The dotted and solid black edges represent celebrity and topic links respectively. Meanwhile, the colored edges represent eight different emotions: red for anger, bright green for joy, dark green for fear, light blue for surprise, dark blue for sadness, pink for disgust, dark orange for anticipation, light yellow for trust. In this example, only two kinds of emotions, i.e., anger and joy, are shown.

6.3 Interface of Constellation Browser.

6.4 Precision of sentiment-oriented clip retrieval on the four opinion questions in Table 6.1.
LIST OF TABLES

3.1 Cele-WebV and its subsets. The second column shows the number of videos, followed by the average number of faces, tagged names and celebrities per video in the remaining columns. The numbers inside parenthesis indicate the percentage of faces without names in the metadata (3rd column) and the percentage of names without faces appearing in the videos (4th column).

3.2 Within-video celebrity naming: Effect of combining multiple relationships. The improvements of CRF-L against UP in accuracy (precision) in “Easy”, “Average”, and “Hard” are 12.9% (22.6%), 31.7% (24.2%) and 38.6% (46.5%) respectively.

3.3 Within-video celebrity naming: performance comparison across different subsets of Cele-WebV.

3.4 Between-video face naming. The improvement of CRF-G against CRF-L is shown in the parenthesis.

4.1 Statistics for the dataset OWE. The second column shows the number of talking tracks, followed by the numbers of sentiment tracks and opinion clips in the third and forth column respectively.

4.2 Accuracy of localizing sentiment-oriented talking tracks. M: video metadata, C: caption.

4.3 Accuracy of the opinion clip localization.

5.1 The number of neurons in each layer of our enhanced multimodal DBM (E-MDBM). $\mathbf{v, h}^1, \mathbf{h}^2, \mathbf{h}^{(v)}$ and $\mathbf{h}^{(J)}$ represent the visible layers, first hidden layers, second hidden layers, joint representation layer over visual features and joint representation layer over visual, auditory and textual modalities.

5.2 Prediction accuracies for each emotion category of VideoEmotion obtained by applying logistic regression to representations learned at different hidden layers. The highest accuracy of each category is highlighted.

5.3 Prediction accuracies of the state-of-the-arts on VideoEmotion [5]. The notations V., Au., and At. represent visual, auditory and attribute features respectively.
5.4 Mean average precision@20 of text-based, video-based and multi-modal query for retrieving emotional videos.

104

5.5 Cross-modal retrieval: Mean average precision @ 20 on four different types of queries against four different versions of datasets.

107

5.6 Cross-modal retrieval: Mean average precision@20 for eight emotion categories under different scenarios. The best performance for each category is highlighted.

107

6.1 The four opinion questions posted of user studies.

118

6.2 The average scores and standard deviation on Precision, Engagement and Acceptance for Constellation and list-wise presentation.

119
CHAPTER 1

INTRODUCTION

1.1 Understanding Perceived Signals

With the massive growth of digital videos in the Internet, recognizing and understanding the visual content is becoming an increasingly important problem. Such techniques have great potential in many important applications, such as exploratory search and open-source intelligence analysis. The problem nevertheless is not trivial especially when considering the wild diversity of user-generated videos in not only content but also shooting environment. Hence, the key difficulty for developing such techniques is the semantic understanding of video content from rich visual-audio signals as well as the sparse and adjective text descriptions. In this chapter, we will first discuss the key signals in the user-generated videos and then describe our proposed methods for understanding these signals and highlight the contributions of this thesis.

By observing the popular videos from YouTube Trends Map\footnote{https://www.youtube.com/trendsmap#sex=0&age=0&type=1}, we find that more than 90 percent of these popular videos are expressing strong emotions, such as the exciting moment in an American football game or the funny moment of dogs. Among these emotional videos, around 80 percent of them are about celebrities, for example, the famous American football player Peyton Manning. We further select a subset of these emotion videos as “sentiment videos”. In an emotion video, the video itself carry a set of signals, which triggers different emotions of the viewers. Meanwhile, the sentimental videos usually only record a person
showing their opinions about specific events or objects. Examples includes Peyton Manning’s talk about the Tom Brady-DeflateGate saga. Hence, in this thesis, we focus on analyzing these three signals, which are **face**, **sentiment** and **emotion**, for understanding the visual content of the user-generated videos.

### 1.1.1 Celebrity Naming

Naming celebrities in the user-generated videos provides great help for both indexing and browsing. To date, most search engines indexing the videos with user-provided text descriptions (e.g., title, tag), which are often noisy and incomplete. In addition, the descriptions are given globally and hence the correspondence between celebrity names and faces is not explicit. Finding the direct correspondences between names and faces could help rectify the potential errors in text descriptions and improve the browsing experience by visualizing this correspondence, for example, by showing the name of a celebrity when a cursor moves over a face.  

The problem of celebrity naming can be traced back to name-face association, where the goal is to align the observed faces with a given set of names. In the literature, this problem has been attempted in the domain of news videos, movies and TV series, capitalizing on the rich set of time-coded information including speech transcripts and subtitles. Nevertheless, these approaches often assume the ideal situation where the text cue is “rich” such that the given name set is free-of-noise and can perfectly match the observed faces. As a consequence, directly extending these approaches to Web video domain is not straightforward. Utilization of rich context information for face naming is also studied in the domain of personal album collection, by using timestamps, geotags, personal contact lists and social networks. Nevertheless, these approaches cannot be directly applied for domain unrestricted videos, because of the absence of context cues and prior knowledge such as family relationships for problem formulation.
1.1.2 Sentiment Analysis for Question Answering

The sentiments of different persons towards specific events or objects are important reference for making decisions, which are usually hotly queried. For example, a query “what’s your opinion” will retrieve 10 millions of questions from Yahoo!Answers. With no surprise, search engines such as YouTube will return more than 2 million hits of videos with this query. Social media has no doubt a platform to voice opinion, and video is becoming a medium for expressing such social activities. Generally speaking, expressing opinion through video has an advantage that the vivid gesture, speaking tone and facial expression are easier comprehended than text-only modality. Despite the advantage and the growth in opinion-related videos, text-only answers remain the major medium because of the grand difficulties in indexing and matching the sentimental video answers, especially when the questions are short in text such as “Why occupy wall street?” To the best of our knowledge, the answering of opinion-oriented questions with videos has not been studied. The goal of answering these opinion-oriented questions is to search a video and locate sentimental segments with potential answers within the video. How to localize the sentimental segments remains a new problem yet to be explored in the literature. As for the matching, traditional way of QA pair matching is by linguistics analysis of sentence structure [14, 15, 16]. Such analysis is not applicable for user-generated video domain as the speech transcripts can be noisy.

1.1.3 Emotion Prediction

Emotion prediction classify the user-generated videos into fine-grained categories than sentimental detection, such as the well-known Plutchik’s wheel of emotions [17] with 24 kinds of emotions. Understanding the perceived emotions inherently
underlying the videos could bright light to emerging applications such as advertising and media analytics. The existing works on emotion prediction of user-generated content are mostly devoted to single media, either text captions or visual contents. Few attempts are paid for combined analysis of multiple media, despite that emotion can be viewed as an expression of multimodal experience. Thus, the learning of highly non-linear relationships exist between different modalities is a critical problem for improving the performance of emotion prediction. In addition, emotion-oriented queries usually not only concern about the emotions but also combine with the semantic objects, such as “dog hates bath”, “angry wedding” and “creepy spider”. Hence, how to combine the emotion prediction with semantic classification posts significant challenge to us, especially when considering the open nature of how nouns (semantic objects) and adjectives (emotions) are combined.

1.1.4 Exploratory Search

Browsing search results while digesting the key information from massive pool of returned data is by no means an easy task. The existing video search engines, which only provide a flat list view, cannot support efficient browsing for complex search queries with multiple semantic facets. In the domain of multimedia particularly, a mechanism that could facilitate the navigation of unexplored results and switch of new search focus is highly demanded. One example of text based exploratory search is the Google “Wonder Wheel”\(^2\). In the system, a wheel shows different semantic facets related to a given query. By rolling different wheels, the users are exposed to diverse aspects of the original query. Similar organizing structure has also been adopted in [18] for multimedia exploratory search. However, the semantic facets used in [18] are limited to news events and the presentation is

\(^2\text{http://www.googlewonderwheel.com/}\)
relatively monotonous. Exploring videos based on signals such as face and emotion are important but not yet addressed for enriching user search experience.

1.2 Overview of Approaches and Contributions

The research in this thesis embraces the goal of multimedia content understanding based on mining the semantic meaning of face, sentiment and emotion signals. We overview our methods and highlight the primary contributions as follows:

- **Unsupervised celebrity face naming in user-generated videos**

  Our main contribution is on the extension of name-face association to domain unrestricted user-generated videos for celebrity face naming. Instead of relying on accurate face labels for supervised learning, a rich set of relationships automatically derived from video content and knowledge from image domain and social cues is leveraged for unsupervised face labeling. The relationships refer to the appearances of faces under different spatio-temporal contexts and their visual similarities. The knowledge includes Web images weakly tagged with celebrity names and the celebrity social networks. The relationships and knowledge are elegantly encoded using conditional random field (CRF) for label inference. Two versions of face annotation are considered: within-video and between-video face labeling. The former addresses the problem of incomplete and noisy labels in metadata, where null assignment of names is allowed – a problem seldom been considered in the literature. The latter further rectifies the errors in metadata, specifically to correct false labels and annotate faces with missing names in the metadata of a video, by considering a group of socially connected videos for joint label inference. Experimental results on a large archive of Web videos show the robustness of the proposed approach in dealing with the problems of missing and false labels, leading to
higher accuracy in face labeling than several existing approaches but with minor degradation in speed efficiency.

- **Opinion question answering by sentiment clip localization**

  We consider multimedia question answering beyond factoid and how-to questions, of particular interest in searching sentiment clips for answering opinion-oriented questions which are controversial and hotly debated. Examples of questions include “Should Edward Snowden be pardoned?” and “Obamacare unconstitutional or not?”. These questions often invoke emotional response, either positively or negatively, and hence are likely to be better answered by videos than texts, due to the vivid display of emotional signals visible through face expression and speaking tone. Nevertheless, a potential answer of duration 60 seconds may be embedded in a video of 10 minutes, resulting in degraded user experience compared to reading answer in text only. Furthermore, a text-based opinion question may be short and vague while the video answers could be verbal, less structured in grammar and noisy because of errors in speech transcription. Direct matching of words or syntactic analysis of sentence structure, such as adopted by factoid and how-to question-answering, is unlikely to find the video answers. The first problem, specifically the answer localization, is addressed by audio-visual analysis of the emotional signals in videos for locating video segments likely expressing opinions. The second problem, questions and answers matching, is tackled by a deep architecture that non-linearly matches text words in questions and speeches in videos. Experiments are conducted on eight controversial topics based on the questions crawled from Yahoo! Answers and the Internet videos from YouTube.

- **Deep multimodal learning for affective analysis and retrieval**
Our main contribution is the proposal of a deep multimodal learning platform that enables a more generalized way of learning features coupled with emotions and semantics. Using Deep Boltzmann Machine (DBM), a joint density model over the space of multimodal inputs, including visual, auditory and textual modalities, is developed. The model is trained directly using user-generated content without any labeling efforts. While the model learns a joint representation over multimodal inputs, training samples in absence of certain modalities can also be leveraged. More importantly, the joint representation enables emotion-oriented cross-modal retrieval, for example, retrieval of videos using text “crazy cat”. The model does not restrict the types of input and output, and hence in principle, emotion prediction and retrieval on any combinations of media are feasible. Extensive experiments on Web videos and images show that the learnt joint representation could be very compact and be complementary to hand-crafted features, leading to performance improvement in both emotion classification and cross-modal retrieval.

• Constellation browser for exploratory search

Empowered by the understanding of perceived signals, an interactive QA system, named Constellation, is developed for exploratory search. The system is built based upon millions of QA pairs for training deep matching between text and speech, about one million of Web image and videos for learning DBM, and three thousand hours of videos for learning word2vec [19, 20] text similarity measure. These deep models, together with CRF for face labeling, are used for creating hyperlinks connecting video clips sharing different perspectives of perceived signals. Our user study indicates that Constellation provides better engagement and user experience than list-wise presentation as most commercial engines adopt.
1.3 Organization

The rest of this thesis is organized as follows. Chapter 2 surveys related works in celebrity face naming, multimedia question answering and affective computation. Chapter 3 presents our CRF based multimodal algorithm for celebrity face naming. In Chapter 4, we present a system for suggesting video-based sentimental answers to opinion-oriented questions. In Chapter 5, we describe our joint representation learning over multimodal inputs for emotion prediction and emotion-oriented queries. Then, an exploratory system is proposed in Chapter 6 based on the analysis of the semantic meaning of these three signals. Finally, Chapter 7 concludes the thesis and discusses our future research directions.

1.4 Publication

The work presented in Chapter 3 was published in IEEE Trans. on Multimedia’15 [21] and the work described in Chapter 4 was accepted in ACM Trans. on Multimedia Computing Communications and Applications’15 [22]. Part of the work discussed in Chapter 5 was accepted in ACM ICMR’15 [23] and an extended version was published in IEEE Trans. on Multimedia’15 [24].

In addition, I have been involved in several other research works during my Ph.D study, including multimodal question answering by naming visual instance (ACM Multimedia’12 [25, 25]) and near duplicate visual content detection for video hyperlinking (ACM Multimedia’11 [18, 26]). The video hyperlinking work was also released as part of an open source project “VIREO-VH” in ACM Multimedia’12 [27].
CHAPTER 2

LITERATURE REVIEW

This chapter surveys the existing research efforts on the three perceived signals – face, sentiment and emotion. Since sentiment analysis is incorporated into a multimedia question answering system, we also discuss the existing works related to multimedia question answering. Furthermore, the research works related to sentiment analysis and emotion prediction are discussed together as “affective computation” because of overlaps in techniques they adopted. Comparisons and contrasts between the existing methods and our proposed works are also drawn out.

2.1 Face Naming

The existing research efforts for face naming are mostly dedicated to the domain of Web images [28, 29, 30] and constrained videos [31, 9] such as TV series, news videos and movies. These works can be broadly categorized into three groups: model-based, search-based and constrained clustering-based face naming.

2.1.1 Model-based Approaches

Model-based approaches seek to learn classifiers for face recognition. Due to the requirement of labeled samples as training examples for each face model, these approaches generally do not scale with the increase number of names. There have been numerous efforts strived for learning effective classifiers from smaller size of
training samples. For example, by using Fisher discriminant analysis the approach in [32] wisely incorporates the labeled and unlabeled samples into kernel learning for face annotation. In [33], partial label information derived from the domain of broadcast videos are exploited for face naming using multiple instance learning. In this case, labeling a face is equivalent to judging whether a face is anonymous or not, which significantly cut short the labeling time. In [34], weakly-labeled images directly crawled from Web are leveraged for learning of face models. To minimize labeling efforts, a bootstrap learning strategy, which is named as consistency learning in [34], is employed to automatically filter out false samples from weakly labeled Web images for model training. A slight deviation of model-based learning is the so-called face verification, which determines whether a face pair belongs to the same person identity. DeepFace [35] is the most recent work achieving great success by using deep learning techniques. However, the requirement of large training samples for adequately covering visual appearance variations, for example, 4.4 million face labels for around 4,000 persons in DeepFace, is resourcefully expensively.

2.1.2 Search-based Approaches

In contrast to model-based learning, search-based approaches mine the names from the retrieved examples deemed to be similar to the query faces. Therefore, the need for training examples is not applicable here since no classifier will be explicitly trained. Generally speaking, the main challenge for this line of approaches is to conquer the problem of noisy labels, for example, by unsupervised label refinement [36], when no supervisory information is available. The problem of name mining is then straightforwardly tackled by majority voting among the top–n retrieved images [36]. In [37], the local coordinate coding (LCC) is applied to enhance the weak labels while minimizing the impact of noisy labels during the voting of top–
2.1.3 Constrained Clustering-based Approaches

The most related works to our work are clustering-based approaches. The underlying assumption is that faces belonging to a person can be densely clustered and hence be exploited for face naming. These approaches generally perform well when there are only a few name candidates to be considered for a face. Existing approaches include constraint Gaussian mixture models (CGMM) [30, 29], graph-based clustering (GC) [29] and face-name association by commute distance (FACD) [39]. Using Expectation-Maximization (EM) algorithm, CGMM [30, 29] learns a Gaussian mixture model for each name. The learning iterates between assigning faces to the best possible model (E-step) and updating of model parameters (M-step). A null category for dealing with missing names problem is also learnt by treating all the faces as a mixture model. GC [29] and FACD [39] adopt a different strategy by using graph representation to model the density of faces. Started from the candidate names given in metadata, GC first retrieves images tagged with these names. A graph is then online constructed with faces in these images as vertices and their similarities as edges. The problem of name assignment is formulated as finding the densest sub-graphs, each corresponding to a name, from the graph. With the constraint that each face in a picture can be assigned to at most one name and vice versa, the problem of name assignment is
shown to be equivalent to min-cost-max-flow problem, which can be solved using simplex algorithm. A merit of GC is that null category assignment can be naturally considered in the problem and no extra parameter is required. FACD strategically speeds up the graph construction by offline indexing the name-face pairs into an inverted index. Different from GC, FACD assigns names by explicitly enumerating the steps, named as commute distance, required to traverse from a face to a name through the random walk algorithm. Compared to GC, an extra threshold needs to learn in order to gate the activation of null category assignment, when the commute distances between a face and the candidate names are considered far. Different from the proposed work in this thesis, these approaches [30, 29, 39, 28] are designed for Web images, and do not exploit inter-image correlation for a more global way of name-face association.

2.2 Multimedia Question Answering

In the literature, multimedia question answering (MQA) is mostly tackled by mining answers from a large volume of multimedia content (e.g., images and videos) on the Web. These works could be broadly classified into three categories based on the type of questions, i.e., “factoid” questions, “how-to” questions and multimodal questions. Furthermore, we also discuss the common methods for filling the lexical and stylistic gaps in question answering systems.

2.2.1 Factoid Question Answering

One of the earliest developed system is VideoQA [40], which leverages visual content, ASR (Automatic Speech Recognition) transcripts and online information for locating news video segments as answers for factoid questions. A passage retrieval algorithm for QA was developed in video documentary domain [41]. By video
caption recognition and pattern-based passage ranking, the algorithm returns the passages associated with short video clips as answers. Following these works, several video QA systems were also proposed to investigate the “factoid” question answering but in different domains, such as educational videos and bilingual videos [42, 43, 44].

2.2.2 How-to Question Answering

In [45], a community-based QA system supporting how-to questions was developed for retrieving Web videos as answers in the domain of consumer electronics. An unified framework for tackling both “factoid” and “how-to” questions was proposed in [46], by extending the text based QA techniques to multimedia QA. The system was designed to find multimedia answers from web-scale media resources such as Flickr and YouTube. More recently, Nie et al. [47] presented a method to predict the media type (text, image, or video) that will best answer the “factoid” and “how-to” questions. Based on the predicted media type, images and videos are retrieved for enriching the text answers.

2.2.3 Multimodal Question Answering

Different from the previous two kinds of questions, which are composed by text, a few research works devoted for multimodal question answering. Specifically, the questions are composed of multimedia objects, such as images and videos, in addition to text. In [48], a QA system was proposed to allow the players in a virtual world to pose questions without textual input. By giving annotated semantic information to the objects in the virtual world, questions and answers are generated based on the contextual information among the objects. However, the system provides very limited questions and answers to the players. Given a
photo as question, photo-based QA [49] exploited visual recognition techniques to answer the factoid questions about physical objects in photos. A more general multimodal QA system was developed in [50] to answer questions of various types, including “factoid”, “how-to” and “opinion” questions. However, the system only provides textual rather than multimedia answers. There are also some commercial websites emerged to provide videos as answers for factoid and how-to questions. The most representative one is eHow\(^1\), which provides how-to videos by recruiting amateur photographers to shoot problem-solving videos. However, producing these videos is cost expensive compared to the automatic search of video answers, as we describe in this thesis.

### 2.2.4 Lexical and Stylistic Gap in QA

Our work is also different from the traditional QA on how to deal with the lexical and stylistic gaps between the question and answer domains. In text QA, these gaps are usually bridged by question reformulation, from rule based rewrites [14], more sophisticated paraphrases [15], to question-to-answer translations [16]. In multimedia QA, the gaps are usually bridged by query expansion [42, 46, 45, 40]. Keywords, which are expanded with contextual related words from WordNet and Web resources, are used for answer matching. In [51], the gaps are bypassed by posting the problem of QA as the similar questions search in community QA websites. The developed technique decomposes the parse tree of a question into tree fragments recursively, and measures the similarity between two questions based on the degree of overlap in tree fragments. This technique only works well for “factoid” and “how-to” questions, but not opinion questions where the answers can vary more wildly in both lexicon and stylistics. In this thesis, inspired by [52], we employ the recent advance in deep learning to bridge the gaps by capturing the

\(^1\)http://www.ehow.com/
localness and hierarchical intrinsics of sentences for question answering.

## 2.3 Affective Computation

Affective computation has been extensively studied in the last decades, and many methods are proposed for handling various media types including textual documents \([53, 54, 55, 56]\), images \([1, 57, 58, 59, 60]\), music \([61, 62, 63]\) and movies \([64, 65, 66, 67]\). Two widely investigated tasks are emotion prediction and sentiment detection. Both of them are standard classification problems with different state spaces. Usually emotion prediction is defined on several discrete emotions, such as anger, sadness, joy etc., while sentiment detection aims at categorizing data into positive or negative. Since the adopted techniques of these two tasks are quite similar, we discuss these two kinds of tasks together in this section. Previous efforts are summarized mainly based on the modality of the data they are working on, i.e., textual documents, images, music and videos.

### 2.3.1 Affection in Textual Documents

For textual data, lexicon-based approach using a set of pre-defined emotional words or icons has been proved to be an effective way. In \([53]\), they propose to predict the sentiment of tweets by using the emoticons (e.g., positive emoticon “:)” and negative one “:-(”) and acronyms (e.g., lol (laugh out loudly), gr8 (great) and rotf (rolling on the floor)). A partial tree kernel is adopted to combine the emoticons, acronyms and Part-of-Speech (POS) tags. In \([55]\), three lexicon emotion dictionaries and POS tags are leveraged to extract linguistic features from the textual documents. In \([54]\), a semantic feature is proposed to address the sparsity of microbloggings. The non-appeared entities are inferred using a pre-defined hierarchical entity structure. For example, “iPad” and “iPhone” indicate the appearance
of “Product/Apple”. Furthermore, the latent sentiment topics are extracted and
the associated sentiment tweets are used to augment the original feature space.
In [56], a set of sentimental aspects, such as opinion strength, emotion and po-
larity indicators, are combined as meta-level features for boosting the sentiment
classification on Twitter messages.

2.3.2 Affection in Images

Affective analysis of images adopts a similar framework with general concept de-
tection. In SentiBank [1], a set of visual concept classifiers, which are strongly
related to emotions and sentiments, are trained based on unlabeled Web images.
Then, a SVM classifier is built upon the output scores of these concept classifiers.
The performance of SentiBank is recently improved by using deep convolution neu-
ral network (CNN) [57]. Nevertheless, the utility of SentiBank is limited by the
number and kind of concepts (or ANPs). Due to the fact that ANPs are visually
emotional concepts, selection of right samples for classifier training could be sub-
jective. In addition to the semantic level features, a set of low-level features, such as
color-histogram and visual aesthetics, are also adopted in [58]. The combined fea-
tures are then fed into a multi-task regression model for emotion prediction. In [4],
hand-crafted features derived from principles-of-art such as balance and harmony
are proposed for recognition of image emotion. In [60], the deep CNN is directly
used for training sentiment classifiers rather than using a mid-level consisting of
some general concepts. Since Web images are weakly labelled, the system progres-
sively select a subset of the training instances with relatively distinct sentiment
labels to reduce the impact of noisy training instances.
2.3.3 Affection in Music

For emotional analysis of music, various hand-crafted features corresponding to different aspects (e.g., melody, timbre and rhythm) of music are proposed. In [62], the early fused features are characterized by cosine radial basis function (RBF). In [68], a ListNet layer is added on top of the RBF layer for ranking the music in valence and arousal in Cartesian coordinates. Besides hand-crafted features, the authors in [63] adopt deep belief networks (DBN) on the Discrete Fourier Transforms (DFTs) of music signals. Then, SVM classifiers are trained on the latent features from hidden layers.

2.3.4 Affection in Video

In the video domain, most research efforts are dedicated to movies. In [64], a large emotional dataset, which contains about 9,800 movie clips, is constructed. SVM classifiers are trained on different low-level features, such as audio features, complexity and color harmony. Then, late fusion is employed to combine the classifiers. In [65], a set of features are proposed based on psychology and cinematography for affective understanding in movies. Early fusion is adopted to combine the extracted features. Other fusion strategies on auditory and visual modalities are studied in [66]. In [67], a hierarchical architecture is proposed for predicting both emotion intensity and emotion types. CRF is adopted to model the temporal information in the video sequence. In addition to movies, a large-scale Web video dataset for emotion analysis is recently proposed in [5], where a simplified multi-kernel SVM is adopted to combine the features from different modalities.
2.3.5 Comparisons

Different from those works, the approach proposed in this thesis is a fully generative model, which defines a joint representation for various features extracted in different modalities. More importantly, the joint representation conveying information from multiple modalities can still be generated when some modalities are missing, which means that our model does not restrict to the media types of user generated contents.
CHAPTER 3

UNSUPERVISED CELEBRITY FACE NAMING IN WEB VIDEOS

Labeling celebrities in Web videos is a challenging problem due to large variations in face appearance. The problem becomes increasingly important due to the massive growth of videos in Internet. According to YouTube trends map\(^1\), about 80 percent of popular videos are people-related and among the people-related videos, about 75 percent are about celebrities. To date, most search engines index these videos with user-provided text descriptions (e.g., title, tag), which are often noisy and incomplete. The descriptions are given globally, and hence the correspondences between celebrity names and faces are not explicit. It is not unusual that a mentioned celebrity does not appear in the video, and vice versa, a celebrity actually appearing in a video is not mentioned. For these reasons, searching people-related videos may yield unsatisfactory retrieval performance, either because of low recall or low precision. Ideally, finding the direct correspondences between names and faces could help rectify the potential errors in text descriptions and thus serve as a preprocessing step for video indexing. Furthermore, user search experience could be improved if the name-face correspondence is visualized, for example, by showing the name of a celebrity when a cursor moves over a face [6].

Figure 3.1 illustrates the problem with a real example of Web video. Out of the fourteen faces (of four celebrities) detected in the video, only four of them have names mentioned in the metadata. Furthermore, among the three celebrities who are mentioned, only two of them appear in the video. In other words, there are

\(^1\)http://www.youtube.com/trendmap
Title: Hillary Clinton and Barack Obama Fight!!!!!!!

Description: During the Democratic presidential debate in South Carolina, Hillary Clinton and Barack Obama engaged in ... past statements on Iraq and refers to a ... about Ronald Reagon, and it was on ...
to a name based on external knowledge from image domain.

- **Face-to-face constraint** considers factors such as background context, spatial overlap, temporal disconnectivity and visual similarity for relating faces from different frames and videos.

- **Name-to-name relationship**, or social relation, considers the joint appearance of celebrities by leveraging social network constructed based on the co-occurrence statistics among celebrities.

The first two relationships are exploited for labeling faces in a video, which we term as “within-video” face labeling. The task is to assign the names mentioned in metadata to the faces detected in a video, with the problem of missing faces and names in mind such that “null assignment” of names is allowed. The social relationship extends naming within a single video to “between-video” naming, by performing labeling of faces on a group of videos whose celebrities fall in the same social network. Compared to “within-video” naming, the relationships established among videos allow the rectification of names incorrectly tagged and the filling in of missing names not found in metadata.

Figure 3.2 depicts two major tasks in this chapter. Given a Web video $V_1$, “within-video” labeling constructs a graph with the names and faces in the video as vertices. Based upon the face-to-name and face-to-face relationships, edges are established among the vertices for inference of face labels by CRF. The inference can be affected by situations such as there are faces whose names are not mentioned in the metadata (e.g., Cenk Uygur), and similarly names mentioned in the metadata but faces do not appear in the video (e.g., Barack Obama). “Between-video” face labeling, by associating $V_1$ to a social network, crawls relevant videos (i.e., $V_2$ and $V_3$) and forms a larger graph composing of names and faces from multiple videos. Using social cues, additional edges modeling name-to-name relationships
Figure 3.2: Within-video naming constructs a graph modeling the face-to-name and face-to-face relationships among the faces and names found in a video \( (V_1) \). By social network, between-video labeling expands the graph by connecting to the graphs of two other videos \( (V_2 \text{ and } V_3) \) that share social relations. The expanded graph is additionally modeled with name-to-name relationship inferred from the social network.

are also established. As shown in the example of Figure 3.2, the expanded graph has the advantages that the missing name “Cenk Uygur” (marked in yellow rectangle) in \( V_1 \) can be propagated from \( V_2 \) and \( V_3 \) and the corresponding faces are assigned with the name replacing the “null” label, while the face wrongly labeled as “Hillary Clinton” (marked in green rectangle) can be rectified with name-to-name relationship as well as the similar faces found in \( V_2 \).

The main contribution of this work is on the extension of name-face association to domain unrestricted Web videos for celebrity face naming. Particularly, this work exploits three major relationships in addressing the problems of missing names and faces commonly happened in weakly-tagged videos, which are issues yet to be fully explored. CRF has been employed in the literature for various labeling tasks, but in different contexts such as image annotation [71] and association of faces and time-coded overlaid text [72], which are different from the proposed
work. We consider CRF in this work mainly for its power in integrating diverse sets of relationships and off-the-shelf algorithms for label inference [73, 74, 75]. CRF, nevertheless, is known to be suffered from slow inference speed and high memory consumption, and hence is prohibited in some applications where scalability is a concern. In this work, we suggest a practical way to bypass this problem by leveraging social relation to constrain the complexity of inference.

This chapter is organized as following. Section 3.1 elaborates the problem formulation and describes our solution based on CRF for within-video celebrity naming, with the consideration of the missing names and faces problem. Section 3.2 extends the solution to between-video celebrity naming, by leveraging social cues to rectify the potential errors in user-provided text descriptions. Section 3.3 presents experimental results, and finally Section 5.6 concludes this chapter.

3.1 Relationship Modeling

This section begins by formulating the problem of within-video face labeling as an optimization problem under conditional random field (CRF). Multiple relationships are then defined to characterize the sets of faces and names in the CRF.

3.1.1 Problem Definition and Notation

Given a video, the inputs to the problem of name-face association are the observed (or detected) faces from the video and the celebrity names found in metadata. Denote the celebrity names as a set $\mathcal{N} = \{c_1, c_2, ..., c_M\}$ and the detected faces as a sequence $S = (x_1, x_2, ..., x_N)$, where $M$ and $N$ represent the number of names and faces respectively. The problem here is to assign at most one name $c_i \in \mathcal{N}$ to a face $x_i \in S$, such that every face in a video is given either a name or no name (i.e., null assignment). The output of the problem is a label sequence, denoted as
Figure 3.3: An example of graph depicting the modeling of relationships for face naming as an optimization problem. The objective function is to maximize the probability of assigning the right names or labels (denoted as $y_i$) to faces based upon the unary and pairwise potentials defined by various relationships.

$Y = (y_1, y_2, ..., y_N)$, where each element $y_i$ is an indexed variable indicating $x_i$ face in the sequence $S$ is assigned with a name $c_i \in \mathcal{N}$ or “null”.

Under CRF, the face and label sequences are modeled as a graph for name inference. The graph is undirected, denoted as $G = (V, E)$, where $V = \{S, Y\}$ is the set of vertices and $E$ is the set of edges connecting vertices. The edges are established based upon different relationships defined between faces and between faces and names. Figure 3.3 shows an example of the graph with 11 faces in the
observed sequence. There is an index variable, $y_i$, encoding the label for each of the eleven faces.

Basically, the problem now is to enumerate each possible label assignment, and then eventually select one among the assignments as the best solution that maximizes the probability of assignment. With a little abuse of notations, let’s denote each of such assignment as a vector $y = [y_1, y_2, \ldots, y_n]$ for a vector of observed faces $x = [x_1, x_2, \ldots, x_n]$. Here, we would like to estimate the conditional probability $p(y|x)$. Following the local Markov property in CRF, we assume that two indexed variables $y_i, y_j \in Y$ are independent of each other if there is no edge (or relationship) between them. With reference to Figure 3.3 as example, the variable $y_1$ is dependent on the variable $y_4$, but not the variable $y_2$. Following the convention of CRF in naming notation [70], we name the set of dependent labels and observations as “factor”. For example, $\{y_1, x_1\}$ is a factor, and $\{y_1, y_4, x_1, x_4\}$ is also a factor. To this end, the conditional probability can be factorized into the following forms

$$p(y|x) = \frac{1}{Z(x)} \prod_{c \in C} \Phi(y_c, x_c)$$

(3.1)

where $C$ is the set of factors in $G$, $\Phi$ is a potential function, and $Z(x) = \sum_y \prod_{c \in C} \Phi(y_c, x_c)$ is a partition function served for normalizing the probability score. We consider two kinds of potentials in characterizing $\Phi$, namely the unary potential $\mu(y_i, x_i)$ and pairwise potential $\psi(y_i, y_j, x_i, x_j)$, which model the face-to-name and face-to-face relationships respectively as shown in Figure 3.3. The conditional probability can thus be rewritten as

$$p(y|x) = \frac{1}{Z(x)} \prod_{c \in C_\mu} \mu(y_i, x_i) \prod_{c \in C_\psi} \psi(y_i, y_j, x_i, x_j)$$

(3.2)

where $C = \{C_\mu, C_\psi\}$. Furthermore, the pairwise potential is a linear combination
of three functions, each features one kind of relationships, as following

$$
\psi(y_i, y_j, x_i, x_j) = \theta_{sr} f_{sr} + \theta_{tr} f_{tr} + \theta_{vr} f_{vr}
$$

(3.3)

Each of these feature functions models spatial ($f_{sr}$), temporal ($f_{tr}$) or visual ($f_{vr}$) relationship, and is weighted by $\theta = \{\theta_{sr}, \theta_{tr}, \theta_{vr}\}$ respectively. In the remaining subsections, we will further detail the unary potential (Section 3.1.2) and the feature functions under pairwise potential (Section 3.1.3).

In brief, the problem of face naming can be elegantly stated as to maximize the probability in Equation 3.1. The inference of names can be rigorously solved with off-the-shelf algorithms such as Markov Chain Monte Carlo (MCMC) [73] or Loopy Belief Propagation (LBP) [74, 75]. As shown in Figure 3.3, the challenges of face naming originate from the large variations in visual appearance and face resolution. Relying merely on face similarity for naming is likely to fail in this kind of examples.

### 3.1.2 Unary Potential

The unary potential energy measures the likelihood of a face $x_i$ being labeled with a name or “null”. To do so, we model each name $n$ as a multivariate Gaussian distribution of faces as

$$
p(x_i^{face}|\lambda_n) = \mathcal{N}(x_i^{face}; \mu_n, \Sigma_n)
$$

(3.4)

where $\lambda = \{\mu, \Sigma\}$ is the set of Gaussian parameters, and $x_i^{face}$ represents the facial feature extracted from face $x_i$. The faces used for modeling Equation 3.4 are extracted from Web images crawled from search engines (See Section 3.3.1.2).

We model the assignment of a face to “null” category as a problem of information uncertainty. Specifically, considering the probability distribution of labeling
a face with the names in $\mathcal{N}$, the uncertainty in labeling can be characterized by the normalized entropy as

$$E_{x_i} = -\sum_{n \in \mathcal{N}} p(x_i^{\text{face}}|\lambda_n) \log_2(p(x_i^{\text{face}}|\lambda_n)) \log_2 |\mathcal{N}|.$$ (3.5)

The uncertainty reaches the highest (i.e., higher entropy value) when the probabilities are uniformly distributed. Reversely, when the probability of assigning to a name is noticeably high than other names, the uncertainty becomes lower. To this end, the unary potential characterizing the edge between a face $x_i$ and a label $y_i$ is defined as

$$\mu(y_i, x_i) = \begin{cases} p(x_i^{\text{face}}|\lambda_{y_i}), & \text{if } y_i \in \mathcal{N} \\ E_{x_i}, & \text{if } y_i = \text{null} \end{cases}$$ (3.6)

where the probability of labeling a face as belonging to “null” category is proportional to the uncertainty of assigning the face to the given names. Note that Equation 3.6 contributes to the conditional probability $p(y|x)$. CRF will numerate all possible name assignments and eventually select the one with the highest probability as the name assignment for $x_i$.

### 3.1.3 Pairwise Potential

The pairwise potential energy characterizes the possible relationships between two faces, as described by Equation 3.3. This sub-section defines the feature functions of each relationship in characterizing the pairwise potential energy.

#### 3.1.3.1 Spatial Relationship

Given two frames of different shots, the spatial locations of faces, as well as their overlapping area, give clue to the identity of face. Generally speaking, by cinematography practice, the position and size of a face shall not change dramatically
across shots for maintaining temporal coherence. Nevertheless, this clue is weak considering that any two faces at the center of frames will overlap regardless of their identities. A more robust way of modeling spatial relationship is to also consider the background of the frames where faces reside. Specifically, when two frames sharing similar background, the spatial relationship can be leveraged to establish edges in $G$ for linking the labels assigned to faces.

Denote $x_i^{\text{back}}$ as a color histogram [76] for the background of a frame where the face $x_i$ resides, the feature function for spatial relation is defined as

$$f_{sr}(y_i, y_j, x_i, x_j) = 1_{\{\cos(x_i^{\text{back}}, x_j^{\text{back}}) \geq \eta\}}$$

(3.7)

The similarity between background frames is measured by cosine similarity, i.e., $\cos(.)$. The parameter $\eta$ is an empirical threshold, which will be discussed in Section 3.3.2. Equation 3.7 specifies the condition for an edge to be established between two faces. Note that the notation $1_{\{\ldots\}}$ is an indicator function which means that an edge between the faces $x_i$ and $x_j$ is established when the condition $\cos(x_i^{\text{back}}, x_j^{\text{back}}) \geq \eta$ is met.

To weight the significance of the knowledge, the parameter $\theta_{sr}$ (Equation 3.3) is considered to be characterized by the positions and sizes of faces. Denote $\text{area}(x_i)$ and $\text{overlap}(x_i, x_j)$ as the size of a face $x_i$ and the area of overlap with face $x_j$ respectively, the proportion of face overlap is

$$\text{prop}(x_i, x_j) = \frac{\text{overlap}(x_i, x_j)}{\text{area}(x_i) \cup \text{area}(x_j)}$$

(3.8)

The value for model parameter $\theta_{sr}$ is set on-the-fly depending on the assigned labels as following

$$\theta_{sr} = \begin{cases} 
\text{prop}(x_i, x_j), & \text{if } y_i = y_j \neq \text{null} \\
1 - \text{prop}(x_i, x_j), & \text{if } y_i \neq y_j \\
\max(\text{prop}(x_i, x_j), 1 - \text{prop}(x_i, x_j)), & \text{if } y_i = y_j = \text{null}
\end{cases}$$

(3.9)
The face overlap is utilized to boost (penalize) when the same (different) names are assigned during label inference. For the case of two faces assigned to null category, the knowledge of whether they belonging to the same identity is unknown. To model the uncertainty, max operator is used such that the assignments will not be punished, regardless of their actual identities.

### 3.1.3.2 Temporal Relationship

The appearance of faces at different timestamps along the temporal axis gives clue of whether the names assigned to faces should be exclusive of each other. Specifically, faces, which coincide in any of a frame along the timeline of a video, should belong to different identities and hence can be assigned different labels throughout the video. Denote $x_i^{time}$ as the timestamp where a face $x_i$ appears, the feature function for temporal relationship is defined as

$$ f_{tr}(y_i, y_j, x_i, x_j) = 1_{\{x_i^{time} = x_j^{time}\}} $$

(3.10)

where an edge between $x_i$ and $x_j$ is established if the condition $x_i^{time} = x_j^{time}$ is fulfilled. Nevertheless, note that this relationship is further controlled by the model parameter $\theta_{tr}$, which can still assign a weight of zero to this relationship if both faces have the same labels, i.e., $y_i = y_j$, as following

$$ \theta_{tr} = \begin{cases} 
1, & \text{if } y_i \neq y_j \text{ or } y_i = y_j = \text{null} \\
0, & \text{if } y_i = y_j \neq \text{null}
\end{cases} $$

(3.11)

A special case is when two faces are both assigned to “null” category. Since this case does not imply that the two faces should belong to the same person, the value for $\theta_{tr}$ is set equal to 1.
3.1.3.3 Visual Relationship

Like spatial relationship, face similarity only provides weak clue to the name identity in the Web video domain. Generally speaking, the dissimilarity between two faces can always be attributed to situations such as changes in viewpoint and lighting conditions. The inference of labels based on face dissimilarity can thus be uncertain. On the other hand, two highly similar faces nevertheless is a necessary clue to evidence the name identity. Based on these two facets of perception on face similarity, the feature function for visual relationship is modeled as

\[ f_{vr}(y_i, y_j, x_i, x_j) = 1_{\{\cos(x_i, x_j) \geq \delta\}}, \]  

(3.12)

where \(\delta\) is an empirical threshold for filtering low similar faces from taking part in the label propagation in CRF. Cosine similarity, \(\cos(x_i, x_j)\), is used for measuring the similarity between two faces \(x_i\) and \(x_j\). The parameter \(\theta_{vr}\) characterized by facial similarity is set as

\[ \theta_{vr} = \begin{cases} 
\cos(x_i, x_j), & \text{if } y_i = y_j \neq \text{null} \\
1 - \cos(x_i, x_j), & \text{if } y_i \neq y_j \\
\max(\cos(x_i, x_j), 1 - \cos(x_i, x_j)), & \text{if } y_i = y_j = \text{null} 
\end{cases} \]  

(3.13)

Similar to Equation 3.9, max operator is used when both labels are assigned to null category.

3.1.4 Complexity Analysis

Algorithm 1 summarizes the major steps of within-video face naming. The runtime complexity is governed by two major parts: graph construction (particularly step-2) and CRF label propagation (step-3). As shown in Algorithm 1, the first two steps involve mainly the establishment of unary and pairwise potentials for the graph \(G(V, E)\), given the sets of faces \(S\) and names \(N\) in a video. For unary
Algorithm 1 Within-video face labeling

**Input:** The sets of faces \( S \) and names \( \mathcal{N} \) in a video \( V \)

**Output:** Face labels \( Y \) that maximizes \( p(y|x) \) in Equation 3.2

1. Constructing a graph \( G \) by modeling the unary potential for each face \( x_i \in S \), where an edge between \( x_i \) and \( y_i \in Y \) is weighted with Equation 3.6.

2. Establishing edges for any pairs of \( y_i \in Y \) and \( y_j \in Y \) in \( G \) that satisfy the condition in Equations 3.7, 3.10 and 3.12, with their edge weights set respectively based on spatial (Equation 3.9), temporal (Equation 3.11) and visual (Equation 3.13) relationships.

3. Performing loopy belief propagation [74, 75] on \( G \) for face labeling.

term, the complexity is \( O(|\mathcal{N}| \times |S|) \), where \( |\mathcal{N}| \) and \( |S| \) are the number of names and faces respectively. Since pairwise potential considers any two face pairs, the complexity is \( O(|S|^2) \). For the third step, we employ Loopy Belief Propagation (LBP) [74, 75] for label propagation. For each face under investigation, the speed of LBP is dominated by message passing, or more specifically, the possible pairwise name assignments \( |\mathcal{N}|^2 \) for every edge of the face. In the worst case when the graph is fully connected, the complexity is \( O(k \times |S| \times |E| \times |\mathcal{N}|^2) \), assuming the inference converges in \( k \) iterations.

The required memory cost is proportional to the size of a graph. Using adjacency matrix as the graph representation, the space complexity is \( O(|S|^2) \). Furthermore, each vertex and edge are represented as a vector of size \( |\mathcal{N}| \) and a matrix of size \( |\mathcal{N}|^2 \) respectively, resulting in extra \( O(|S| \times |\mathcal{N}|) \) and \( O(|E| \times |\mathcal{N}|^2) \) for storing vertices and edges. In practice, as the spatial and visual relationships connect only faces with high background and visual similarities respectively, the constructed graph is generally sparse. As a consequence, the complexity in terms of speed and space for a video is not considered high.
Figure 3.4: Between-video naming: the process of constructing social networks and performing CRF on multiple smaller graphs.

### 3.2 Leveraging Social Cues

Performing face naming within a video has the limitation that only the names mentioned in the metadata will be considered. In the situation where missing names or faces exist, probabilistic inference of labels could become arbitrary due to lack of sufficient clues. In the extreme case, there maybe no edges established for some faces after the evaluation of pairwise relationships. Under this situation, the amount of messages passing between faces will be limited, which directly impacts the effect of label propagation. By extending the proposed approach in Section 3.1 to beyond a single video, faces originally lacking channels for effective message passing should have higher chance to be connected. To be explicit, the advantages of involving multiple videos in graph construction are twofold. By the candidate names from third party videos, faces originally labeled as null can be named as far as possible through CRF optimization. Second, ambiguous labels due to information uncertainty can be resolved with additional cues derived from other videos.
3.2.1 Social-driven Graph Splitting

With a video collection as input, the graph $G$ presented in Section 3.1.1 is expanded with a vertex set $V$ including all the observed faces and names mentioned in the collection. The expansion will result in dramatic increase of edges under the modeling of unary and pairwise potentials. Specifically, unary potential considers all the available names as admissible labels for a face, while visual relationships establish links for similar faces across videos. Note that the spatial and temporal relationships, which are only valid within a video itself, cannot be leveraged for between-video connectivity. Due to the quadratic complexity of CRF, the significant increase in size of a graph is expected to affect the speed efficiency. As an example in the experiment, considering a collection of 2,000 videos, processing each video sequentially will take 27 minutes in total. Jointly processing the videos as a whole will slow down the speed by 200 times theoretically.

A vivid cue that can be exploited for reducing the size of graph is social networks among the celebrities under consideration. For example, the former president of China Jintao Hu is not likely to be linked to any face in a video about “Britain’s Got Talent show”. In other words, social network helps trimming potentially superfluous relations, giving light of splitting a large graph into subgraphs of each depicting a social network. With this intuition, we first exploit the co-occurrence of celebrities in mining social networks, followed by constructing one graph per social network for CRF optimization. Figure 3.4 summarizes the name inference using social networks, where there are three major steps.

Denote $G_{name} = (V, E, W)$ as a graph depicting the relationship among celebrities, where $V$ is the vertex set representing $n$ names, and $E$ is the edge set linking celebrities. The notation $W$ denotes a weight matrix of size $n \times n$, whose element
$W_{i,j}$ describes the relationship between two celebrities $n_i$ and $n_j$ defined as

$$W_{i,j} = \frac{2|s_i \cap s_j|}{|s_i| + |s_j|}$$  \hspace{1cm} (3.14)$$

where $s_i$ is the set of videos tagged with celebrity $n_i$, and $|s_i|$ denotes the set cardinality. Equation 3.14 basically calculates the co-occurrence statistics of celebrities as the proportion of videos where both names are tagged. As can be seen in the step-1 of Figure 3.4, using the matrix $W$, a social graph $G_{name}$ is constructed. We connect each name in $G_{name}$ to five other names with the largest weights, such that the resulting graph is sparse and efficient to be processed. Subsequently, by Walktrap algorithm [77], the graph is further partitioned into sub-graphs corresponding to social networks. The algorithm is highly efficient and capable of estimating the number of communities automatically.

Having the social networks, we distribute each video to one or multiple networks based on the names mentioned in a video (step-2 of Figure 3.4). Denote $\Omega_C$ as the set of celebrities in a community $C$, and $N_{v_i}$ as the set of names tagged in a video $v_i$. A video $v_i$ is assigned to $C$ if $N_{v_i} \cap \Omega_C \neq \emptyset$. By doing so, each network $C$ will be associated with a video pool $V_C$, and meanwhile the community size will also be expanded with new names from the pool, i.e., $\Omega^*_C = \Omega_C \cup \bigcup_{v_i \in V_C} N_{v_i}$. In other words, $\Omega^*_C$ is composed of celebrities not only from the network but also all names tagged in the videos assigned to $C$. To this end, as depicted in step-3 of Figure 3.4, CRF only needs to separately consider the videos and celebrities $\Omega^*$ in the domain of a community for name inferencing, which overall cuts short the running time.

### 3.2.2 Name-to-name Social Relation

Recall that the joint modeling of videos as a graph has the advantage that a name $y_j$ not mentioned in a video $v_i$ can still be exploited for labeling faces in
Intuitively, the chance that there is a celebrity named $y_j$ appearing in $v_i$ is proportional to the co-occurrence of $y_j$ with other names mentioned in $v_i$. For example, Bill Clinton has a higher chance than George Bush to appear in a video tagged with the name Barack Obama, giving the fact that Clinton has closer political relationship with Obama. With this intuition, the unary potential term in Equation 3.6, which characterizes the edge between a face $x_i$ and a label $y_i$, can be augmented with social information, or name-to-name relationship, as following

$$
\mu(y_i, x_i) = \begin{cases} 
w_{y_i,x_i}, & \text{if } y_i \in N \\
E_{x_i}, & \text{if } y_i = \text{null}
\end{cases}
$$

where

$$
w_{y_i,x_i} = \alpha \times p(x_i^{\text{face}}|\lambda_{y_i}) + (1 - \alpha) \times \max(W_{n,y_i}), n \in N
$$

and $W$ is the matrix capturing the co-occurrence statistics of celebrities. The max operator basically picks a name $n \in N_v$ who co-occurs most with $y_i$, and uses the corresponding score in $W$ for adjusting the significance of unary potential. Basically, the equation takes into account whether the label $y_i$ is actually socially connected to any names mentioned in a video $v_i$. The probability of being $y_i$ is boosted, if the statistics supports such claim. Equation 3.16 combines both clues from visual and social in a linear fashion. The empirical parameter $\alpha$, which is set equal to 0.5, is for trading off the importance of visual and social cues.

### 3.2.3 Algorithm Implementation

While running CRF separately on each community as shown in Figure 3.4 has accelerated face naming, a practical concern is the memory cost. For a community with 13 celebrities, 279 videos (i.e., $|V_C| = 279$) and 34 candidate names (i.e., $|\Omega_C| = 34$), the memory consumption can be as high as 5G bytes. Furthermore, for videos with celebrities who could be assigned to more than one community,
the result of labeling may not be consistent across communities. For practical consideration, we thus propose CRF to be conducted on video rather than on community basis.

Algorithm 2 summarizes the details. The algorithm starts by processing each video individually. Given a video $v_i$, step-1 constructs a graph $G_i$ by establishing the face-to-name and face-to-face relationships based on the unary and pairwise potentials. Step-2 crawls videos sharing the same social networks as $v_i$. Subsequently, step-3 builds a new graph $G^*$ to establish edges among the involved videos based on face similarity and name-to-name relationship. Among the crawled videos, $G^*$ eventually only includes videos that establish edges with $v_i$, which is usually a sparse and small graph in size. For example, there are typically about 30 videos per graph in our experiments. Performing face inference on such graph, step-4 of the algorithm, will take about one second and occupy only 0.04G bytes of memory.

**Algorithm 2** Between-video face labeling

**Input:** A Web video $v_i$ and the associated metadata  
**Output:** Face labels

1. Construct a graph $G_i$ connecting faces and names in $v_i$ by equations 3.4, 3.7, 3.10 and 3.12.

2. Crawl the videos, and their graphs, sharing the same social networks as $v_i$ from the dataset.

3. By equations 3.12 and 3.15, a new graph $G^*$ is constructed by establishing edges to connect all the graphs in steps 1 and 2.

4. Perform loopy belief propagation [74, 75] on $G^*$ for face labeling.

In Algorithm 2, it is important to note that steps 1 to 3 only need to be performed once for all the videos in a dataset. Precisely, with reference to the
Table 3.1: Cele-WebV and its subsets. The second column shows the number of videos, followed by the average number of faces, tagged names and celebrities per video in the remaining columns. The numbers inside parenthesis indicate the percentage of faces without names in the metadata (3rd column) and the percentage of names without faces appearing in the videos (4th column).

<table>
<thead>
<tr>
<th></th>
<th>video</th>
<th>face</th>
<th>name</th>
<th>celebrity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Easy</td>
<td>2223</td>
<td>15.0 (54%)</td>
<td>1.2 (46%)</td>
<td>0.65</td>
</tr>
<tr>
<td>Average</td>
<td>257</td>
<td>20.0 (52%)</td>
<td>3.4 (59%)</td>
<td>1.35</td>
</tr>
<tr>
<td>Hard</td>
<td>103</td>
<td>25.8 (59%)</td>
<td>5.9 (67%)</td>
<td>1.96</td>
</tr>
<tr>
<td>Cele-WebV</td>
<td>2583</td>
<td>15.9 (54%)</td>
<td>1.6 (52%)</td>
<td>0.77</td>
</tr>
</tbody>
</table>

social networks, multiple large graphs can be constructed for all the videos in a dataset. When labeling faces for a video $v_i$, only videos which connect to $v_i$ will be involved during face inference. Algorithm 2 can also be directly applied for unseen or newly arrived videos. Basically, by crawling all the videos sharing the same social networks as an unseen video (step-2), a graph $G^*$ can be built (step-3) on-the-fly by relationship modeling among these videos for name inference (step-4).

### 3.3 Experiments

The experiments include within-video (Section 3.3.3) and between-video (Section 3.3.4) face naming. The empirical studies investigate the effectiveness of various relations proposed in this chapter, with comparison to state-of-the-art approaches. The runtime efficiency is also detailed in Section 3.3.5.
3.3.1 Dataset and Evaluation Metrics

3.3.1.1 Dataset

A dataset named Cele-WebV [78] is constructed for experiments. The dataset is originated from the core dataset of MCG-WEBV [79], composing of 14,473 Web videos uploaded to 15 YouTube channels during December of year 2008 to November of year 2009. To preprocess the dataset, candidate person names are extracted from the video metadata, by stepwise matching of a word as well as a succession of words against Wikipedia. A person name is verified if the category tag for birth year is found in the matched Wikipedia pages. By filtering out names that appear in less than 10 videos, finally there are 141 celebrity names being retained for experiments. The dataset Cele-WebV is formed by pooling together 2,583 videos containing the 141 celebrities. A total of 41,047 frontal faces are extracted\(^2\) from 409,900 keyframes of the dataset, including 20% of close-up faces with resolution larger than 150 \(\times\) 150 pixels.

To facilitate the result analysis, we further split the dataset into three subsets: \textit{Easy}, \textit{Average} and \textit{Hard}, representing the potential difficulty in face naming. The \textit{Easy} (\textit{Hard}) subset contains videos with no more than 2 (more than 4) celebrity names found in the metadata. The \textit{Average} dataset includes the remaining videos with 3 or 4 names. Table 3.1 shows the detailed statistics of Cele-WebV dataset.

To generate ground-truth, each face is labeled with a celebrity name found in the video metadata. By doing so, each celebrity has on average 136.5 faces. However, there are only 46% (19,240 out of 41,047) of faces being labeled with names. On average, each video has 8.56 faces without assigning a name. On the other hand, there are 52% of celebrity names do not associate with any faces in the videos. The large number of faces without a name, as well as names without corresponding

\(^2\)We employ the commercial software developed by ISVision for face detection: http://www.isvision.com/cn/index
We also create another subset named Cele-WebV* by fully labeling all the faces in a video regardless of whether the celebrity names are mentioned in the metadata of the video. There are 300 videos randomly selected from Cele-WebV being included in this subset. Different from Cele-WebV, we expand the number of celebrities from 141 to 200 names. Out of the 2,487 faces without names found in the metadata, 146 faces are labeled with a name among the 200 celebrities. Finally, we create another dataset named Cele-WebV+ consisting of 800 videos not in the core dataset of MCG-WEBV. Note that there is no overlap of videos between Cele-WebV+ and Cele-WebV, though both sets of videos are originated from MCG-WEBV. As Cele-WebV*, the 7,663 faces in the dataset are fully labeled with the names of 200 celebrities. Among 3,739 faces (48.8%) without celebrity names in the metadata, 223 of them are labeled. Compared with Cele-WebV, there is a higher percentage of names, on average around 1.27 out of 2.1 names in the metadata, without any faces found in the video.

3.3.1.2 Supporting Image Dataset and Features

A face dataset consisting of Web images of the 200 celebrities is also constructed. The pictures are crawled from Google image search engine using the celebrity names as the keywords. The top-150 pictures presented by the search engine for each celebrity are crawled. No human intervention is involved throughout the procedure. To reduce noises, pictures with more than one faces are filtered out. Finally, there is a total of 19,851 images in the dataset. The dataset is used for modeling the unary potential as described in Section 3.1.2. For each celebrity, a multivariate Gaussian model is learnt, by treating the Web images of the celebrity as positive examples. Note that the relatively small number of Web images, around 100 per celebrity, hinders the use of more complex model such Gaussian mixture
model (GMM).

Two features, facial feature and color histogram, are extracted for measuring the face and background similarities respectively. The facial feature is represented by a 1,937 dimensional vector, describing the salient points extracted from 13 facial regions [9]. The dimension of the facial feature is reduced to 100 by principal component analysis (PCA). The background feature is represented by a color histogram of 300 dimensions in RGB space. Cosine similarity is employed as proximity measurement for background features [76].

3.3.1.3 Performance Evaluation

Similar to [39, 31, 28, 29], the performance is measured by accuracy and precision. Both measures count the number of faces correctly labeled, but differ where accuracy also includes the counting of faces without labels. Denote \( C \) as the set of faces correctly labeled (including “null assignment”), and \( T \) as the groundtruth labels of all the faces in a dataset. The set \( T = \{L, U\} \), which is composed of a subset \( L \) with all the faces assigned with names and a subset \( U \) with faces without assigned a name. The definitions of accuracy and precision are:

\[
\text{Accuracy} = \frac{|C \cap T|}{|T|} \quad \text{(3.17a)}
\]

\[
\text{Precision} = \frac{|C \cap L|}{|L|} \quad \text{(3.17b)}
\]

Note that accuracy and precision are calculated across all the faces in a test collection, rather than averaged over videos. Recall is not used here because we do not consider the problem of “retrieving all faces given a name”, rather we are dealing with the problem of whether a face is labeled with a correct name (precision), and otherwise labeled as “null” if the name is missing from metadata (accuracy).

There are three datasets used in the experiments. The experiments on within-
Table 3.2: Within-video celebrity naming: Effect of combining multiple relationships. The improvements of CRF-L against UP in accuracy (precision) in “Easy”, “Average”, and “Hard” are 12.9% (22.6%), 31.7% (24.2%) and 38.6% (46.5%) respectively.

<table>
<thead>
<tr>
<th>Relation</th>
<th>Accuracy</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Easy</td>
<td>Average</td>
</tr>
<tr>
<td>CRF-L</td>
<td>0.603</td>
<td>0.532</td>
</tr>
<tr>
<td>UP+Temporal (TR)</td>
<td>0.573</td>
<td>0.437</td>
</tr>
<tr>
<td>UP+Visual (VR)</td>
<td>0.555</td>
<td>0.422</td>
</tr>
<tr>
<td>UP+Spatial (SR)</td>
<td>0.576</td>
<td>0.425</td>
</tr>
<tr>
<td>Unary Potential (UP)</td>
<td>0.534</td>
<td>0.404</td>
</tr>
</tbody>
</table>

Figure 3.5: Sensitivity of the parameters $\eta$ (left) and $\delta$ (right) towards the accuracy and precision of face naming.

video and between-video face naming are conducted on Cele-WebV and Cele-WebV* respectively. Cele-WebV is also used for empirical parameter tuning. The parameters of the proposed as well as compared approaches are tuned using grid search in a brute-force manner for the best possible performance. For more objective evaluation, the dataset Cele-WebV$^+$ is used as an independent dataset for evaluation, where no parameter tuning is allowed.

3.3.2 Parameter Sensitivity

There are two empirical parameters, $\eta$ and $\delta$, used in our method for controlling the graph sparsity. These parameters are for filtering out the faces with low facial and background similarities, as outlined in the equations 3.7 and 3.12 respectively.
Figure 3.5 shows the sensitivity of these two parameters on Cele-WebV. Both parameters exhibit similar performance trends, where a lower value will result in inclusion of noises, and hence lower performances in both accuracy and precision. On the other hand, an extremely high value will filter most of the true positive relationships. The performances drop in this case since there are only few edges left to be leveraged for multiple relationships modeling. Basically, for both parameters the performances peak at certain values, which are relatively high to filter most noises while still capable of retaining a good number of true positives for estimation. With a reasonable setting, the performances vary within the ranges of 46% - 52% for precision, and 50% - 58% for accuracy. In the remaining sections, the parameters of $\eta$ and $\delta$ are set to 0.9 and 0.85 respectively.

### 3.3.3 Within-video Face Naming

We first investigate the effect of considering multiple relationships in our approach (CRF-L). Table 3.2 contrasts the performances when different relationships are incorporated. On top of the unary potential (UP) defined based upon external Web images, each relationship basically improves the accuracy (precision) by at least 7% (5%). Visual relationship is observed to be contributing less than other relationships mainly due to the large variation of face appearance in Web videos, attributed to the uncontrolled video capturing environment. By integrating all the three relationships, an improvement of 16% (18%) is attained in accuracy (precision). It is also observed that the performance is inversely proportional to the number of names mentioned in metadata and the number of faces in a video. It is worth noting that, when multiple relationships are leveraged, the relative improvement is indeed proportional to the number of faces and names. Generally speaking, the large number of faces and names increases the uncertainty of naming, but results in a graph with more relationships to be exploited on the
Table 3.3: Within-video celebrity naming: performance comparison across different subsets of Cele-WebV.

<table>
<thead>
<tr>
<th></th>
<th>Accuracy</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Easy</td>
<td>Average</td>
</tr>
<tr>
<td>RA</td>
<td>0.471</td>
<td>0.232</td>
</tr>
<tr>
<td>TA</td>
<td>0.536</td>
<td>0.360</td>
</tr>
<tr>
<td>CGMM [30, 29]</td>
<td>0.535</td>
<td>0.403</td>
</tr>
<tr>
<td>GC [29]</td>
<td>0.545</td>
<td>0.419</td>
</tr>
<tr>
<td>FACD [39]</td>
<td>0.590</td>
<td>0.457</td>
</tr>
<tr>
<td>CRF-L</td>
<td>0.603</td>
<td>0.532</td>
</tr>
</tbody>
</table>

other hand. For example, the appearance of multiple faces in a frame hints the exclusive relationships when assigning the names. As shown in Table 3.2, the temporal relationship contributes more to the performance in the Hard subset than the Average and Easy subsets. It is worthwhile to note that the accuracy of null assignment (because of missing names) is also improved when more names are given in the metadata. For Hard subset, the accuracy is 52.3%, against the 47.9% and 48.5% achieved by the Average and Easy subsets respectively. This is simply because the entropy (Equation 3.5) used for measuring the uncertainty of name-face assignment becomes more reliable, when more names are available for providing a more complete picture of statistics. Similar observation is also noted for the missing faces problem. Lower error rate (31.6%) is attained in Hard subset than Average (37.2%) and Easy (35.8%), due to the presence of more relationships to be exploited for inference.

Next, we compare our proposed method CRF-L with five other approaches: random assignment (RA), threshold-based null assignment (TA), constrained Gaussian Mixture Model (CGMM) [30, 29], graph-based clustering (GC) [29] and commute distance (FACD) [39]. RA is a baseline, which randomly associates a face either to a name in the metadata or to null category. TA is based on our method but considering only the unary potential, and an empirical threshold optimally tuned for null assignment. Note that CGMM, GC and FACD are originally de-
signed for name-face association in the domain of Web images. In the experiments, similar to CRF-L, these approaches operate on the keyframe-level and directly use the names tagged in metadata as the candidate names for labeling. Both CGMM and GC involve iterative optimization. In the experiment, the iteration stops as soon as no more than 3% of face labels are updated. The learning process normally converges within 10 iterations. For FACD, there are two key parameters: top-\( p \) similar faces of a name to be used for graph construction, and the threshold \( \epsilon \) for null category assignment. We tune the parameters in the range of values suggested by [39], and choose \( p=50 \), and \( \epsilon \) equals to a value where 40% of faces in the dataset are assumed belonging to null category based on commute distance. For all the three approaches, note that temporal relationship is also considered and utilized as the “cannot link” constraint for restricting the assignment of names.

Table 3.3 lists the comparison of six different approaches. The general trend is that all the approaches outperform the baselines RA and TA, and the performance gaps become wider with the increase difficulty level of the dataset. CRF-L exhibits the overall best performance across all the subsets. Using the unary potential and temporal relation alone (UP+TR) as shown in Table 3.2, the accuracy (0.536) is already higher than all other compared approaches. A key observation is that facial similarity alone is not always reliable in Web video domain, where the appearance can be wildly different even within the same video. As a consequence, CGMM, GC and FACD, which depends heavily on facial similarity for model learning and graph construction, suffer from imprecise modeling. Although temporal relation is also considered by these approaches, the fact that the relation is utilized as a hard constraint, rather than soft constraint as in CRF-L, also limits its power in label estimation. In short, CRF-L enjoys the advantages that multiple relations, in addition to facial similarity, can softly interact to reach consensus, resulting in more robust face labeling than other approaches. By integrating multiple relation-
Table 3.4: Between-video face naming. The improvement of CRF-G against CRF-L is shown in the parenthesis.

<table>
<thead>
<tr>
<th></th>
<th>Accuracy</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>UP</td>
<td>0.480</td>
<td>0.395</td>
</tr>
<tr>
<td>CL</td>
<td>0.483</td>
<td>0.401</td>
</tr>
<tr>
<td>CRF-L</td>
<td>0.561</td>
<td>0.503</td>
</tr>
<tr>
<td>CRF-G</td>
<td>0.586 (+4.5%)</td>
<td>0.521 (+3.6%)</td>
</tr>
</tbody>
</table>

ships softly, CRF-L can also effectively minimize the error propagation in message passing.

3.3.4 Between-video Face Naming

This section verifies the performance of our approach (CRF-G), which models multiple relationships not only within videos but also among videos. The experiment is conducted on Cele-WebV*. For CRF-G, the 200 celebrities are split into 12 communities using Walktrap algorithm [77]. The community size ranges from as small as 15 videos with 4 celebrities to as large as 81 videos with 28 celebrities. Depending on which communities the tagged celebrities belong to, each video in Cele-WebV is assigned to one or multiple communities. As presented in Section 3.2.3, CRF-G builds an extended graph for each video based on the connections of a video with other videos in the communities. We compare CRF-G to consistency learning (CL) [34], which label faces regardless of whether names are tagged. CL builds a face model per name using images crawled from Web. In the implementation, we use the face dataset crawled from Google (see Section 3.3.1.2) as training examples for learning 200 k-nearest-neighbor (k-NN) classifiers as the face models. Here, the value of k is empirically tuned to 5, which exhibits the best performance when tested across different values of k. As in [34], the bootstrapping strategy is employed for selecting the best possible samples for learning k-NN. By doing so,
there are 2,831 Web images in the face dataset being filtered out by CL. The motivation of CL can be viewed as similar to the unary potential energy in CRF-G, except with a more sophisticated way of sample selection and classifier learning.

Table 3.4 shows the performance comparison. Recall that in Cele-WebV*, out of 2,487 faces not being tagged, there are 146 faces belonging to one of the 200 celebrities. By CRF-G, there are 79 out of these 146 faces being correctly named. Furthermore, there are also 687 incorrectly labeled faces by CRF-L being rectified, where among them 55 of the labels are due to the missing faces problem. On the other hand, CRF-G does generate false alarms, where 345 faces, correctly assigned with “null”, are incorrectly labeled, and 51 faces, correctly named by CRF-L, are falsely labeled. The false labels are mostly attributed to facial features, which adjust the uncertainty of labeling (Equation 3.5) for “null” assignment. Overall, CRF-G outperforms CRF-L by 4.5% and 3.6% respectively in terms of accuracy and precision. Comparing to CL and UP, it is apparent that leveraging multiple relations has advantage over classifier learning. Similar to the observation as in within-face naming, visual information alone suffers from imprecise estimation due to wildly different appearances of faces across videos.
3.3.5 Performance Comparison

This section shows the experimental results on Cele-WebV+, where no parameter tuning is allowed. Performance comparison is conducted for a total of eight different approaches listed in Figure 3.6. As shown in the results, CRF-G shows the overall best performances in accuracy and precision followed by CRF-L. The performance trend is similar to that of observed on Cele-WebV and Cele-Web* datasets. To verify that the performance of different methods presented in Figure 3.6 is not by chance, we also conduct significance test using randomization test [80]. The target number of iterations used in the randomization is 100,000. At the significance level of 0.05, CRF-G significantly outperforms all other approaches including CRF-L. Meanwhile, the performance of CRF-L is also significantly better than all other six compared approaches.

Figure 3.7 details the online processing time of six different approaches on Cele-WebV using a PC with 8-core 2.67GHZ cpu and 20GB memory. Note that the decomposition of celebrities into communities (in CRF-G), model learning (in CL), and construction of inverted index (in FACD) are all considered offline and the time for these operations are not shown. For CRF, online processing includes the time spent for graph construction and face naming. In CRF-L, the average size of a graph is 16 vertices (faces), 19 edges (pairwise potential) and 3.1 labels (candidate names and null label). Considering that a total of 27 minutes is required for processing a video collection of 880 hours, CRF-L is fairly efficient. When introducing external knowledge from other videos and considering celebrity relationships by social networks, the average graph size is grown to 84 vertices, 113 edges and 12.3 labels. Additional 40% of time is required for CRF-G compared to CRF-L. CGMM and GC, in contrast to CRF-G and CRF-L which process each video individually, consider all the videos and labels in one run, resulting
in slower speed. Among all the approaches, CL is the most efficient in terms of online processing. The efficiency, however, is traded off by the expensive offline processing in sampling training examples for classifier learning.

3.3.6 Discussion

This subsection further discusses the factors that could impact the performance and practicality of CRF-G.

Impact of social relation. We examine the impact of social cue in face naming, by comparing CRF-G to the case when social relation is not considered. Precisely, CRF is run on a graph constructed using all the videos in Cele-WebV*, and without social relation to split the graph and adjust the unary potential (Equation 3.15). The experimental result shows that social relation speeds up CRF-G by 16 times, from 3,809 seconds to 241 seconds. In terms of performance effectiveness, social relation contributes 20% and 29% of improvements for the accuracy and precision of CRF-G respectively. The results basically indicate that running CRF-G on multiple smaller sub-graphs will not degrade the effectiveness, and meanwhile, significantly speed up the efficiency.
Figure 3.8: Samples of “noisy” (bottom) versus “clean” (top) faces, due to different effects: pose, illumination, resolution, make-up, occlusion, aging, drawing (from left to right).

Scalability. To study the scalability of CRF-G, we further expand Cele-WebV* from 200 to 1,000 celebrities. This results in a relatively large social network, which is split into 87 communities after applying Walktrap algorithm on the network. The accuracy and precision attained for this expanded dataset are 0.556 and 0.508 respectively. The results correspond to the drops of 5.1% in accuracy and 2.5% in precision when comparing to running CRF-G on 200 celebrities. As CRF-G operates on multiple smaller communities rather the whole social network, the increase in the number of celebrities basically means more number of communities to be processed, but does not necessarily imply that the scale (in terms of the number of faces and names) of each community will also expand proportionally. As a result, the overall performance is not impacted adversely.

Variations in face appearances. Next, we investigate the robustness of CRF-G to face variations. Among the 4,564 faces in Cele-WebV*, there are 3,441 faces, or 75% of faces, being manually picked up and regarded as suffering from changes in pose, illumination and resolution. Figure 3.8 shows some samples of clean and noisy faces. Labeling this subset of faces is generally challenging, for example, using unary potential alone can only attain the accuracy of 0.404 and precision of 0.260. CRF-G is able to achieve the accuracy of 0.514 and precision of 0.404, which basically show the benefit of exploiting different relationships for this problem.
Figure 3.9: True and false positives by CRF-G: (a) the face $f_3$ is strongly connected to the label “Barack Obama”, and indirectly influences the labeling of three other faces suffering from changes in face appearances; (b) two visually similar faces ($f_1$: Alexandra Burke, $f_2$: Beyonce Knowles) result in false labeling of $f_2$.

When clean and noisy faces are linked in a graph, CRF-G possesses the capability in propagating the strong beliefs from clean to relatively noisy faces, which is the key reason leading to performance improvement. Figure 3.9(a) shows an example illustrating how noisy faces can be correctly labeled. On the other hand, when two faces are incorrectly linked, as an example shown in Figure 3.9(b), false labeling is also likely to happen. Comparing to the other subset of clean faces (accuracy = 0.806, precision = 0.784), the performance drops due to face variations are 36% in accuracy and 48% in precision. Basically, the success in labeling depends mostly on whether the right and correct relationships are established among the clean and noisy faces for message passing by CRF-G.

In modeling unary potential, we employ the multivariate Gaussian with single distribution for modeling unary potential. Considering that Gaussian mixture model (GMM) has better capability in capturing face variations, we also conduct additional experiment investigating the advantage of GMM. Using the expanded Cele-WebV* dataset with 1,000 celebrities for the experiment, a total of 96,314
Figure 3.10: The tradeoff between t-precision and t-recall when thresholding the results of CRF-G on Cele-WebV*.

Web images for these celebrities were crawled. We assume that the images of a celebrity contain ten Gaussian components. Using GMM, the accuracy (precision) is boosted to 0.569 (0.511), corresponding to 2.9% (1.9%) of improvement over the approach that does not employ GMM. Nevertheless, our analysis shows that GMM helps very little in rectifying errors due to severe face variations. We speculate that, due to the different data distributions in image and video domains, the effectiveness of GMM is limited though helpful in capturing variations peculiar to the video domain, such as the effects of resolution and occlusion shown in Figure 3.8.

**Practicality.** The CRF-G formulation is to maximize the conditional probability $p(y|x)$ as in Equation 3.1. In the application where it is better not to suggest the name of a celebrity than assigning an incorrect name, we can set a threshold to gate whether a name should be assigned to a face given the value of $p(y|x)$. With this intuition, we set a threshold $t$ such that a face $x_i$ is assigned to a name indexed by $y_i$ only if $p(y_i|x_i) \geq t$. Figure 3.10 shows the tradeoff between precision and recall by thresholding on $p(y|x)$. Note that t-precision is defined as the number of correctly labeled faces over the number of labeled faces given the threshold set as $t$. Similarly, t-recall measures the proportion of faces being correctly labeled out of all the faces with labels in the dataset. “Null assignment” is not considered here.
because the assignment means not to assign a name to a face. From Figure 3.10, it can be seen that setting the threshold to the value of 0.15 can achieve a precision of 0.843, while still with a recall of 0.4. Beyond this threshold point, precision continues improving slightly but recall tends to drop significantly.

3.4 Summary

We have presented the modeling of multiple relationships using CRF for celebrity naming in the Web video domain. In view of the incomplete and noisy metadata, CRF softly encodes these relationships while allowing null assignments by considering the uncertainty in labeling. Experimental results basically show that these nice properties lead to performance superiority over several existing approaches. The consideration of between-video relationships also results in further performance boost, mostly attributed to the capability of rectifying the errors due to missing names and persons. The price of improvement, nevertheless, also comes along with increase in processing time and the number of false positives. Fortunately, the proposals of leveraging social relation and joint labeling by sequential video processing still make CRF scalable in terms of speed and memory efficiency. Currently the social networks “are” mined based on the co-occurrence of the celebrities in our dataset. A more objective way is by constructing the social networks based on some public “datasets”, such as Internet Movie Database\(^3\), which could potentially provide a more holistic view of celebrity relationships.

\(^3\)http://www.imdb.com
CHAPTER 4

OPINION QUESTION ANSWERING WITH SENTIMENTAL CLIPS

This chapter addresses the problem of matching opinion-oriented text questions to video answers. Figure 4.1 illustrates the problem with an example of question “Opinions on Chick-fil-A against gay?” with only a few words. The goal is to search a video and locate the segment with potential answers to the question. As observed from the speech transcripts, there are very few words in the target segment matching to the question. On the other hand, the keywords “Chick-fil-A” and “gay” are distributed throughout the videos, making the chance of locating the right segment of answer very slim. We address the challenge in three steps: preprocess the videos by analyzing the sentiment content (Section 4.1), localize the opinion-oriented segments based on audio-visual cues (Section 4.2), and perform nonlinear matching of speech tracks with the text question posted by user (Section 4.3). The novelty of this three-step process is originated from narrowing the search scope of video answers by integrating non-textual evidences for opinion clip localization, and the proposal of a deep learning architecture for matching text questions and the speech tracks of video clips.

Figure 4.2 depicts the proposed framework with three major building blocks corresponding to the aforementioned three-step process. The first building block decomposes input videos into segments by speaker diarization. A series of content processing, including face tracking, speech transcription and caption extraction, is performed on each segment. The extracted multi-modal signals, with the aid
Question: Opinions on Chick-fil-A against gay?

Video: Chick-Fil-A President Shoots Anti-Gay Remarks

The Chick-Fil-A restaurant chain is known for more on this let’s turn to democratic strategist Keith Boykin and syndicated radio host Michael Medved. Michael Medved, do you think Chick-Fil-A is promoting institutional bigotry against gay marriage?

No, not at all, I think the bigotry that is being shown as the bigotry against Chick-Fil-A. Mr. Cathy, the president of the company, has only said that he supports traditional marriage. He supports biblical marriage. They didn’t discriminate their gay employees.

... in terms of Chick-Fil-A argument. Boston Mayor is promising to black Chick-Fil-A from... Chick-Fil-A doesn’t belong...

Figure 4.1: An example of opinion question and video answer. In this chapter, we are interested in locating a segment in the video (the highlighted box) that has an answer to the question. The challenges include locating a clip where an opinion holder expresses views of the question from a lengthy video, and the fact that there are very few overlapping words between text question and speech transcripts (underlined) for reliable matching.

of an ontology for sentiment inference, are further analyzed to identify the sentiment talking tracks while filtering the non-sentimental-oriented segments, which are mostly for information rather than opinion expression. The second building block aims to locate the talking tracks with opinion holders, which are referred to as opinion clips in this chapter. We differentiate the opinion holders from subjects such as anchor persons and journalists, whose roles are to deliver information or moderate discussion than voicing personal opinion. To do so, a variety of ad-hoc features obtained through audio-visual processing of sentiment tracks are derived for characterizing opinion holders. A heuristic reasoning algorithm based on expectation-maximization algorithm is then proposed for the selection of opinion clips. Finally, given a question, the third building block searches and ranks the opinion clips in database that match the question. The key component for matching is a deep neural network designed and learned specifically for mod-
Figure 4.2: Framework of the proposed system. The boxes in green color indicate potential video segments to be selected for question answering.

eling the latent semantics of topics of interest. The network enables the nonlinear matching of short texts and video speeches through latent semantics, which is potentially more powerful than the traditional ways of keyword matching such as TF-IDF.

We emphasize that this chapter addresses only the opinion questions with sentiment tendencies. Answering questions such as “Why are the uniforms in the Olympics for every country in English?” is out of the scope. But questions such as “Are U.S. doctors like Obamacare? If yes or no, why?” which are likely to invoke emotional responses and answers, rather than absolute answers, fall into the interest of this chapter. Furthermore, this chapter targets for finding sentiment clips with opinion holders, rather than clips showing emotional signals but no opinion holders, as candidate answers. Only in such clips where emotion is visible through the expressions or gestures of opinion holders, the advantage of using videos as answers to controversial topics can be demonstrated. Based on this
assumption, only those videos with on-camera opinion holders are considered in this chapter. Concretely, opinion question is defined as the question that does not have absolute answers and is likely to invoke sentiment discussion. The associated answers are restricted to the opinions expressed directly by the opinion holders rather than third-parties. The contributions of this work can be summarized as follows:

- Opinion QA: To the best of our knowledge, the answering of opinion questions with videos has not been studied. Previous works on multimedia QA fall in the categories of answering “factoid” and “how-to” questions, which could be tackled by directly matching the texts observed in videos and questions [45, 46, 40]. Opinion QA poses challenges to these works because there could be very few or even no overlap in words between a question and an answer. For example, the best answer chosen in Yahoo! Answers for the question “Why did Romney insult the NAACP (National Association for the Advancement of Colored People)?” is “It was deliberate: he tried to pander to the GOP base of white supremacists & Christian ultra-nationalists”, where there is no word intersection.

- Clip localization: Answering by not only providing a video, but also the clips that likely to answer a question, remains a new problem yet to be explored in the literature. This work gives light of how possibly answer localization can be done in the domain of sentiment-based QA.

- Cross-media matching: Traditional way of QA pair matching is by linguistics analysis of sentence structure [14, 15, 16]. Such analysis is not applicable for video domain as the speech transcripts can be noisy. This chapter proposes the employment of a deep neutral network, which is learnt by QA pairs in text domain, but is leveraged for matching video answers. This again is a
new technique not previously attempted.

The application scenario of the proposed work is to retrieve and rank videos which could answer opinion questions. Particularly, the locations where potential answers reside in a video can be made known to facilitate video browsing. Furthermore, video segments with opinion holders who deliver stronger emotion signals are preferred when presenting the potential answers. The remaining of this chapter is organized as following. Section 4.1 presents the preprocessing step, which includes the extraction, classification and filtering of talking tracks. Only sentiment tracks are retained after the preprocessing step. Section 4.2 further describes an unsupervised learning algorithm for locating opinion clips out of the sentiment talking tracks. Section 4.3 presents the architecture of deep neural network in matching the text-based questions and opinion clips. The techniques for topic modeling and parameter learning are described. Finally, Section 4.4 presents experimental results, and Section 4.5 summarizes this chapter.

4.1 Sentiment Detection

This section outlines the method for detecting the sentiment-oriented speeches in video domain. We start by presenting the extraction of talking tracks (Section 4.1.1), followed by classification of the talking tracks based on their sentiment content (Section 4.1.2). The major challenge is that speech analysis alone, obtained through ASR, is imperfect for sentiment analysis. We approach this problem by considering multiple sources of modalities for analysis.

4.1.1 Extracting Talking Tracks

We employ speaker diarization for video partitioning [81]. Different from the definition of shot, each partition corresponds to a segment with a person speaking.
The technique performs hierarchical agglomerative clustering of speakers by Cross-likelihood Ratio (CLR) on the audio track. No prior information, such as the number of speakers or samples of voices, are required.

Given the video partitions where each has a speaker identity, we are only interested in the segments with talking heads. Voice-over segments, which generally deliver only the background information of a topic, are excluded. To do this, the speaker face in a segment has to be detected and tracked\(^1\). We employ Viola-Jones detector [82] for detecting the frontal faces, followed by Kanade-Lucas-Tomasi (KLT) method [83] for tracking the faces across frames. Face tracks extracted in such a way were shown to be robust to occlusion and drift problem as demonstrated in [84]. To determine whether a person is talking, the mouth region has to be tracked also. We utilize the dark area inside a mouth region as cue for detection. A face track is declared as belonging to a talking person, if there is a consecutive and significant change in the proportion of dark area to the size of mouth area. To ensure the robustness of tracking, a series of steps is carried out, including normalizing face to a canonical pose with a resolution of 80 × 80 pixels based on the position of eyes, and performing histogram equalization to diminish the sensitivity to skin colors.

4.1.2 Sentiment Classification

A simple way of identifying whether the speech content of a talking track is sentimental is by spotting the words such as support and repeal from the speech. However, this method performs poorly in practice. Instead, we extract three features from speeches, captions and metadata respectively from a track, and then develop Naive Bayes classifier for sentiment detection. Each feature is rep-

\(^1\)Note that using audio for talking head detection is not necessary because speaker diarization basically ensures that each segment contains only one speaker.
resented as a binary vector of 155,287 dimensions, where each dimension refers to a word in SentiWordNet [85]. An element of the vector is set to the value of 1 if the corresponding word is present. Denote \( \Upsilon \in \{A, C, M\} \) as a binary vector of \( N \) dimensions for speeches (A), captions (C) and metadata (M) respectively, we can approximate the probability distribution of a feature given a sentiment \( s \in \{\text{positivity, negativity, neutrality}\} \) as

\[
P(\Upsilon \mid s) = \prod_{j=1}^{N} P(w_j \mid s)^{n_j(\Upsilon)}, \tag{4.1}
\]

assuming words are conditionally independent. The function \( n_j(\Upsilon) \) outputs 1 if word \( w_j \) presents and 0 otherwise. \( P(w_j \mid s) \) is estimated by SentiWordNet, which outputs a value in the range of 0 to 1 indicating the degree of sentiment.

By Bayes’ rule, the sentiment \( s \) of a talking track \( T \) is defined as

\[
P(s \mid T) = \frac{P(s)P(T \mid s)}{P(T)}, \tag{4.2}
\]

where \( Z \) is a normalizing constant that can be omitted. Because metadata provides prior regarding the sentiment of a video, we estimate \( P(s) \) with \( P(M \mid s) \) in Equation 4.1. Furthermore, \( P(T \mid s) \) is jointly estimated by \( P(A \mid s) \) and \( P(C \mid s) \). To this end, we develop Naive Bayes classifier as

\[
P(s \mid T) \sim P(M \mid s)P(A \mid s)P(C \mid s) \tag{4.3a}
\]

\[
\sim \prod_{j=1}^{N} P(w_j \mid s)^{n_j(M)+n_j(A)+n_j(C)}, \tag{4.3b}
\]

We want to emphasize that, by using SentiWordNet, no training data is required for classifier learning. Based on Equation 4.3(b), we use log-likelihood to estimate the sentiment score as following:

\[
L(s \mid T) = \sum_{j=1}^{N} \log P(w_j \mid s)^{n_j(M)+n_j(A)+n_j(C)} \tag{4.4}
\]
To this end, each talking track is associated with three sentiment scores. The tracks with higher scores in neutrality than positivity and negativity are filtered out from further processing. Note that in addition to SentiWordNet, there are other methods for sentiment analysis such as [86, 87, 88], which operate directly on image features. Nevertheless, these works cannot be directly applied to our problem. This is mainly because we consider only video clips with human subjects as the focuses and hence the sentiment signals are mostly from spoken content and surrounding texts rather than visual effect.

4.2 Locate Opinion Clip

In this section, we are interested in locating the video segments that have opinion holders expressing views for a topic of interest. Under our problem definition, opinion clip localization is a task equivalent to the selection of sentiment-oriented talking tracks from opinion holders. Section 4.2.1 presents the ad-hoc features derived for characterizing the opinion holder. Based on these features, expectation-maximization (EM) algorithm is proposed for opinion clip localization (Section 4.2.2).

4.2.1 Features

We adopt a heuristic approach for the identification of opinion holders. First, the duration of delivering opinion, indicated by the length of a talking track, shall be relatively longer than the ones from non-opinion holders such as the host or anchor person. Second, an opinion holder should possess higher sentiment score as computed by Equation 4.4. Third, the name of opinion holder is often shown in the video caption. In contrast, a non-opinion holder tends to speak in a relatively shorter duration, appears at the beginning and end of a talk show, and mentions
frequently the name of opinion holder. Sometimes, there are voice-over segments introducing the background history of a topic by a non-opinion holder. To vividly translate these heuristics into numeric scores, a total of 11 features based on audio-visual cue processing are extracted for representing a talking track. These features are briefly described as followings.

- **Visual appearance frequency** \( (f_1) \). Face diarization [89], which groups face tracks based on visual similarity, is performed to cluster the talking tracks in a video. The similarity is measured based upon the set of facial feature points extracted from frontal faces. For a given talking track \( T_i \), the feature \( f_1 \) counts the percentage of face tracks in a video that falls into the same cluster as \( T_i \).

- **Audio appearance frequency** \( (f_2) \). Based on the result of speaker diarization in Section 4.1.1, each talking track \( T_i \) is associated with a speaker cluster. Similar to \( f_1 \) but based on audio processing, the feature \( f_2 \) measures the percentage of talking tracks falling into the same speaker cluster as \( T_i \).

- **Voice-over** \( (f_3) \). Video segments with voice but without face appearance are regarded as voice-over segments. The feature \( f_3 \) of a talking track \( T_i \) is assigned to the value of 1 if there exists a voice-over segment that falls into the same speaker cluster as \( T_i \), and 0 otherwise.

- **Duration** \( (f_4) \). The duration of a talking track.

- **Temporal location** \( (f_5) \). A discrete value, of value ranges from 1 to 3, is assigned depending on whether a talking track appears at the beginning (1), middle (2) or end (3) of a video sequence. The beginning (end) is defined as the first (last) 5% of video length.
• **Number of person names** \((f_6, f_7, f_8)\). Named-entity recognition is performed on the ASR transcript. The feature \(f_6\) measures the number of distinct person names that are mentioned in the speech of a talking person. The feature \(f_7\) \((f_8)\), on the other hand, counts the number of distinct names detected \(n\) second before \((\text{after})\) the speech of a talking person. A non-opinion holder is expected to possess higher value of \(f_6\) for introducing opinion holder(s). In reverse, an opinion holder is expected to possess higher values of \(f_7\) and \(f_8\) for introduced by a host before the start and after the end of his or her speech.

• **Number of names in subtitles** \((f_9)\). The name of opinion holder is assumed appearing in subtitles together with the opinion holder. The feature \(f_9\) counts the number of distinct names found in the captions and subtitles along with a talking track.

• **Number of faces** \((f_{10})\). A typical studio setup for interview is that all the faces of hosts and opinion holders are visible at the beginning, followed by the middle or close-up shots of each opinion holder when expressing opinion. The feature \(f_{10}\) counts the number of face tracks appearing together with a talking track \(T_i\), where ideally the value should be lower for a talking track belonging to an opinion holder.

• **Sentiment score** \((f_{11})\). The feature \(f_{11}\) is a score representing the degree of positivity or negativity in sentiment as computed by Equation 4.4.

Ideally, an opinion holder takes longer time to express opinions (higher \(f_4\) value) than non-opinion holders, while the frequency of an opinion holder appearing in a video is usually lower (lower \(f_1\) and \(f_2\) values). He or she should express opinion for a controversial topic (higher value of \(f_{11}\)), probably supplemented with visual
cues such as their names being introduced by host (higher $f_7$ value) and printed on screen (higher $f_9$ value). In addition, clips with only one opinion holder in the scene (lower $f_{10}$ value) is more focus and thus preferred.

4.2.2 EM algorithm

Based on the eleven designed features, we adopt expectation maximization (EM) algorithm for clustering the talking tracks into the categories of opinion and non-opinion holders. Assuming these features are conditionally independent, the probability of a talking track $T_i \in T$ given a category $c_j$ is:

$$p(T_i | c_j) = \prod_{k=1}^{K} p(t_{i,k} | c_j)$$

(4.5)

where $t_{i,k}$ denotes the the $k$th feature of $T_i$, and $K = 11$ is the length of feature vector. We model the features with continuous value ($f_1, f_2, f_4, f_{11}$) using normal distribution, and the features with discrete value using multinomial distribution. The model parameters $\theta$ are estimated by maximizing the log-likelihood of joint distribution in E step:

$$L(T; \theta) = \log(\prod_{i=1}^{N} p(T_i | \theta)) = \sum_{i=1}^{N} \log(\sum_{k=1}^{K} p(c_j) p(t_{i,k} | c_j, \theta))$$

(4.6)

where $N$ is the number of talking tracks in $T$. In M step, the posterior probability of $c_j$ is updated by Bayes’ rule:

$$p(c_j | T_i)^{new} = \frac{p(c_j)^{old} p(T_i | c_j)^{old}}{p(c_1)^{old} p(T_i | c_1)^{old} + p(c_2)^{old} p(T_i | c_2)^{old}}$$

(4.7)

The parameters of both normal and multinomial distributions for each category $c_j$ are updated separately. For features modeled with normal distribution, we
update the mean $\mu_{j,k}^{\text{new}}$ and variance $\sigma_{j,k}^{\text{new}}$ with the following equations:

$$
\mu_{j,k}^{\text{new}} = \frac{\sum_{i=1}^{N} p(c_j \mid T_i)^{\text{new}} t_{i,k}}{\sum_{i=1}^{N} p(c_j \mid T_i)^{\text{new}}} \quad (4.8)
$$

$$
\sigma_{j,k}^{\text{new}} = \frac{\sum_{i=1}^{N} p(c_j \mid T_i)^{\text{new}} (t_{i,k} - \mu_{j,k}^{\text{new}})^2}{\sum_{i=1}^{N} p(c_j \mid T_i)^{\text{new}}} \quad (4.9)
$$

For multinomial distribution, the marginal probabilities over features are directly updated as following:

$$
p(t_{i,k} \mid c_j)^{\text{new}} = \prod_{x=1}^{X} [p(t_{i,k} = x \mid c_j)^{\text{new}}] 1(t_{i,k} = x), \quad (4.10)
$$

where $X$ is the number of possible values in feature $t_{i,k}$ and $1(\cdot)$ is an indicator function. The probability $p(t_{i,k} = x \mid c_j)^{\text{new}}$ is further smoothed as following:

$$
p(t_{i,k} = x \mid c_j)^{\text{new}} = \frac{1 + \sum_{r=1}^{N} p(c_j \mid T_r)^{\text{new}} 1(t_{r,k} = x)}{X + \sum_{r=1}^{N} p(c_j \mid T_r)^{\text{new}}}, \quad (4.11)
$$

To this end, the category model is updated as following:

$$
p(c_j)^{\text{new}} \approx \frac{1}{N} \sum_{i=1}^{N} p(c_j \mid T_i)^{\text{new}} \quad (4.12)
$$

$$
p(T_i \mid c_j)^{\text{new}} = \prod_{k=1}^{K} p(T_{i,k} \mid c_j)^{\text{new}} \quad (4.13)
$$

In the implementation, we employ K-means to estimate the initial parameters. E-step and M-step are iterated until convergence. Finally, the cluster with the lower value of $f_{10}$ is selected, and the corresponding talking tracks are regarded as belonging to the opinion holders. The heuristics is practical because non-opinion holders often appear together with one or several speakers, and the frequency of appearance is usually higher. This strategy also works well for personal videos, which usually have one person talking throughout a video.
4.3 Question-Answering by Deep Learning

Next, we describe the matching of text question with candidate video answers, more specifically the extracted opinion clips as presented in the previous section. Generally speaking, question-answering is by no means an easy task because of the lexical and stylistic gaps in how questions are asked and answers are elaborated. Lexical gap refers to the vocabulary difference between questions and answers, while stylistic gap refers to the syntactic difference in sentence structure. The gaps lead to ambiguity in word features. Traditional relevance measures based on frequency of overlapping words, such as Cosine similarity and KL-divergence, are not effective for question-answer semantic modeling. Inspired by the ideas in DeepMatch [52], which models the relevance between questions and answers with deep neural network, we construct a new deep architecture (named as DeepHPam) with hierarchical Pachinko Allocation Model (hPam) [90] to make composite decision on matching opinion clips to questions in a hierarchical way (Section 4.3.1). The major improvement of DeepHPam over DeepMatch is that we incorporate the probability distribution over the latent semantics of topics, learnt by hPam, into the construction (Section 4.3.2) and initialization (Section 4.3.3) of the hierarchical structure of neural network.

4.3.1 Topic Modeling by hPam

We view question-answering as a translation problem, where questions and answers are treated as two separate domains. The task is to compute how likely an answer can be “translated” from a question of a different domain. To achieve this, two vocabularies, one from each domain, are generated. Denote $|V_q|$ and $|V_a|$ as the sizes of question and answer vocabularies. A QA pair is represented by a vector of length $|V_q| + |V_a|$, by concatenating both vocabularies. An element in the vector
encodes the frequency of a word. Using the vectors as input, the algorithm hPam [90] captures the salient patterns composed of words from different domains. For example, for the topic “Obamacare”, the pattern (or event) “job” is described by two sets of words, “kill, job, unemployed ...” and “trim, company, fire ...”, in the question and answer domains respectively.

With hPam, a layered directed acyclic graph (DAG) is constructed to model the events under a topic. Here, we refer “topic” as a subject of discussion, such as “Obamacare” and “Edward Snowden”. An event is regarded as a “latent subtopic” mined by hPam for characterizing topic generation. DAG organizes the major events of a topic into a two-level hierarchical graph. Figure 4.3 shows an example of DAG constructed for the topic “Obamacare”, where each node represents an event encoded by a set of words. For example, the event “grass root” is composed of words such as “bankrupt” and “dole”. Furthermore, each word is associated with a probability indicating its likelihood to an event. The two-level hierarchy models the event granularity, describing how an event at higher layer is generated from a mixture of events at lower level. For example, the event “effect” at Layer-II
can be jointly modeled with the events “job” and “income” in Layer-I. By DAG, a question or answer can be represented by “latent subtopics” or events. For example, the question “What might the effects of Obamacare be on jobs for lower middle citizens?” is jointly described by “job”, “income” and “middle class”, by using the DAG shown in Figure 4.3. An advantage of using this representation is that a potential answer to a question does not necessarily to have overlap in words, as long as sharing similar event distributions.

Note that each event in DAG is composed of words from both domains. To suppress noise, for each event in Layer-I of DAG, we only pick the top-10 words with the highest probabilities from the question domain, and the top-20 words from the answer domain for encoding a question-answer (QA) pair. The reason for picking more words from the answer domain is due to the fact that the length of an answer is used to be longer than a question. To this end, with respect to an event, a QA pair is represented as a vector of 10 × 20 dimensions, where each dimension corresponds to a word pair composed of words in the question and answer domains. We use binary vector representation in this case. Specifically, an element in the vector is set to a value of 1 if the corresponding word pair is observed in QA pair. The vector is then fed into neural network for learning and classification.

### 4.3.2 Deep Architecture

The architecture of neural network is depicted in Figure 4.4. The first two hidden layers, p-layerI and p-layerII, correspond to the first and second layers of a DAG, where each event is jointly modeled by three neurons. In the implementation of hPam, we learn 333 and 100 events for the first and second layers respectively. As a result, p-layerI and p-layerII in the deep architecture are composed of 999 and
300 neurons correspondingly. The input to the neural network is composed of 333 binary vectors, of each $10 \times 20$ dimensions generated by one of the events in DAG. The connection between the input layer and p-layerI is based on the event where a binary vector belongs to. In other words, for a binary vector of an event, the $10 \times 20$ elements are fully connected to the three neurons in charge of this event, but have no connection with any other neurons.

The neurons between p-layerI and p-layerII are partially connected, simulating the hierarchy of DAG. A committee layer of 20 neurons is further added and fully connected with p-layerII to model the event relationships at higher semantic level. Finally, the output layer generates a matching score indicating the goodness of translation from question to answer. For all the neurons, we adopt sigmoid function as the activation function.
4.3.3 Hyper Parameter Learning

In the architecture, there are totally 506,840 parameters to be learnt and only around 40% of them are kept to be nonzero values. The parameters between playerI and playerII are initialized based on the probability distribution learnt by hPam in modeling events between the two levels of hierarchy. Similarly, the parameters between input-layer and playerI are initialized based on the outcome of hPam. Specifically, based on Bayes’ rule, we multiply the probabilities of two words in a pair as the initial value for the parameter connecting a word pair and a neuron. The parameter values are updated and learnt by employing a discriminative training strategy with a large margin objective. The training instance is in the form of triple \((x, y^+, y^-)\), where \(x\) is a question, \(y^+\) is the corresponding answer, and \(y^-\) is a false answer. We define the following ranking-based loss as objective:

\[
\mathcal{L}(W, D_{trn}) = \sum_{(x_i, y^+_i, y^-_i) \in D_{trn}} e_W(x_i, y^+_i, y^-_i) + R(W),
\]

where \(R(W)\) is the L2 regularization term, and \(e_W(x_i, y^+_i, y^-_i)\) is the error for a triple \((x_i, y^+_i, y^-_i)\), given by the following large margin form:

\[
e_W(x_i, y^+_i, y^-_i) = \max(0, m + s(x_i, y^-_i) - s(x_i, y^+_i)),
\]

with \(s(x, y)\) represents the score of output layer and \(0 < m < 1\) controls the margin in training. Here, we empirically set \(m = 0.1\). We use stochastic sub-gradient descent with mini-batches [91] for training, where each batch consists of 20 randomly generated triples \((x, y^+, y^-)\).

4.4 Experiments

The experiments are split into three parts for evaluating the accuracy of opinion clip localization (Section 4.4.2), opinion question answering (Section 4.4.3) and a
Table 4.1: Statistics for the dataset OWE. The second column shows the number of talking tracks, followed by the numbers of sentiment tracks and opinion clips in the third and forth column respectively.

<table>
<thead>
<tr>
<th>Topics</th>
<th>talking tracks</th>
<th>#sentiment tracks</th>
<th>#opinion clips</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obamacare</td>
<td>1082</td>
<td>979</td>
<td>904</td>
</tr>
<tr>
<td>Syria</td>
<td>1536</td>
<td>1381</td>
<td>993</td>
</tr>
<tr>
<td>Snowden</td>
<td>1339</td>
<td>1116</td>
<td>871</td>
</tr>
<tr>
<td>Shutdown</td>
<td>1332</td>
<td>1253</td>
<td>1001</td>
</tr>
<tr>
<td>Tax Return</td>
<td>957</td>
<td>738</td>
<td>683</td>
</tr>
<tr>
<td>Chick-fil-A</td>
<td>1294</td>
<td>1037</td>
<td>697</td>
</tr>
<tr>
<td>Occupy W.S.</td>
<td>1810</td>
<td>1206</td>
<td>1099</td>
</tr>
<tr>
<td>NAACP Speech</td>
<td>981</td>
<td>913</td>
<td>873</td>
</tr>
<tr>
<td>All</td>
<td>10331</td>
<td>8623</td>
<td>7121</td>
</tr>
</tbody>
</table>

user study (Section 4.4.4) supporting the claim in this chapter.

4.4.1 Datasets

4.4.1.1 Video Dataset

A Web video dataset named OWE (Opinion Web vidEo) was constructed for experimentation. The dataset is composed of eight opinion-oriented topics, consisting of 800 videos with duration around 340 hours. The eight topics are “Affordable Care Act” (Obamacare), “Syria chemical weapons” (Syria), “Edward Snowden” (Snowden), “US government shutdown 2013” (Shutdown), “Mitt Romney’s Tax Return” (Tax Return), “Chick-fil-A same-sex marriage controversy” (Chick-fil-A), “Occupy Wall Street” (Occupy W.S.), and “Romney’s speech to NAACP” (NAACP Speech). These topics are highly controversial and have triggered many discussions in Yahoo! Answers and thus are selected for experimentation. In the dataset, about 26% of the videos are personal videos, and the remains are official news videos, such as talk show. Table 4.1 shows the details of OWE dataset.
To construct the dataset, the top-100 ranked videos of each topic along with their metadata and ASR\textsuperscript{2} were downloaded from YouTube. The toolset Tesseract\textsuperscript{3} was employed to extract captions and subtitles from the videos. The talking tracks, which were extracted from the videos based on the method in Section 4.1.1, were manually labeled by human evaluators. Three human evaluators, who are familiar with the eight topics, were recruited for ground-truth generation. The evaluators were asked to first label whether a talking track is sentiment-oriented based on speech content. The evaluators were further instructed to judge whether a sentiment-oriented track contains an opinion holder expressing the view for a topic of interest. During this process, we guaranteed that each talking track will be evaluated by at least two evaluators. Any inconsistency in labeling will be picked up and judged by the third evaluator. Table 4.1 shows the statistics of OWE dataset, where there are around 7,000 opinion clips out of about 10,000 talking tracks being annotated in the dataset.

4.4.1.2 QA Dataset

Another corpus composed of question-answer pairs from Yahoo! Answers was constructed for the eight topics, using the same keywords posted to YouTube. There is a total of 53,611 QA pairs and each question has 7.6 answers on average. Based on our observation, most of these questions are opinion questions or opinion-related. The QA pairs are pre-processed by stopword removal and stemming. The average vocabulary sizes for questions and answers are 18,630 and 36,896 respectively. There are 42,000 QA pairs generated for each topic on average.

To learn the network parameters, we sampled 31,000 $\langle x, y^+, y^- \rangle$ triples from

\textsuperscript{2}As personal videos were often captured in indoor environment, the results of speech recognition are acceptable for question-answering. Note that some ASR of personal videos are actually transcripts uploaded by the video owners.

\textsuperscript{3}https://code.google.com/p/tesseract-ocr/
Table 4.2: Accuracy of localizing sentiment-oriented talking tracks. M: video metadata, C: caption.

<table>
<thead>
<tr>
<th>Topics</th>
<th>ASR</th>
<th>ASR+C</th>
<th>ASR+M</th>
<th>ASR+M+C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obamacare</td>
<td>0.681</td>
<td>0.693</td>
<td>0.697</td>
<td>0.735</td>
</tr>
<tr>
<td>Syria</td>
<td>0.705</td>
<td>0.729</td>
<td>0.763</td>
<td>0.797</td>
</tr>
<tr>
<td>Snowden</td>
<td>0.678</td>
<td>0.711</td>
<td>0.754</td>
<td>0.810</td>
</tr>
<tr>
<td>Shutdown</td>
<td>0.677</td>
<td>0.696</td>
<td>0.711</td>
<td>0.746</td>
</tr>
<tr>
<td>Tax Return</td>
<td>0.690</td>
<td>0.721</td>
<td>0.749</td>
<td>0.773</td>
</tr>
<tr>
<td>Chick-fil-A</td>
<td>0.591</td>
<td>0.613</td>
<td>0.621</td>
<td>0.655</td>
</tr>
<tr>
<td>Occupy W.S.</td>
<td>0.578</td>
<td>0.593</td>
<td>0.619</td>
<td>0.622</td>
</tr>
<tr>
<td>NAACP Speech</td>
<td>0.697</td>
<td>0.751</td>
<td>0.739</td>
<td>0.790</td>
</tr>
<tr>
<td>All</td>
<td>0.657</td>
<td>0.682</td>
<td>0.702</td>
<td><strong>0.734</strong></td>
</tr>
</tbody>
</table>

The collected QA pairs for each topic. The answer \( y^+ \) was selected from either the best answer picked by an asker or any answer given by a user. The false answer \( y^- \) was randomly selected from the answer pool of a topic. In learning the neural network, a random subset of 1,000 triples was picked as validation set for parameter tuning, including setting the coefficients for L2 regularization.

The testing set was formed by randomly picking ten questions per topic from the QA corpus. The performance is measured by nDCG@10 with three levels (2, 1, 0) of relevance, where 2 means a retrieved opinion clip fully answering a question, while 1 means partial answer and 0 means not relevant to a question. For example, for the question “Opinions on Chick-fil-A against gays?”, the opinion clip “No, not at all, I think the bigotry that is being shown as the bigotry against Chick-fil-A. Mr. Cathy, the president of the company, has only said that he supports traditional marriage. He supports biblical marriage ... They didn’t discriminate their gay employees.” was labeled as 2. Another clip “It’s fascism for these guys to say you cannot come to my town if I disagree with your political view. That’s fascism.” was labeled as 1, since the speech is about the comments of Boston’s mayor “Anti-gay Chick-fil-A not welcome in this city”. The relevance score for the clip “Work
Table 4.3: Accuracy of the opinion clip localization.

<table>
<thead>
<tr>
<th>Topics</th>
<th>Random</th>
<th>K-Means</th>
<th>EM Learning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obamacare</td>
<td>0.287</td>
<td>0.513</td>
<td><strong>0.631</strong></td>
</tr>
<tr>
<td>Syria</td>
<td>0.391</td>
<td>0.505</td>
<td><strong>0.619</strong></td>
</tr>
<tr>
<td>Snowden</td>
<td>0.357</td>
<td>0.492</td>
<td><strong>0.597</strong></td>
</tr>
<tr>
<td>Shutdown</td>
<td>0.311</td>
<td>0.501</td>
<td><strong>0.573</strong></td>
</tr>
<tr>
<td>Tax Return</td>
<td>0.306</td>
<td>0.487</td>
<td><strong>0.502</strong></td>
</tr>
<tr>
<td>Chick-fil-A</td>
<td>0.332</td>
<td>0.476</td>
<td><strong>0.514</strong></td>
</tr>
<tr>
<td>Occupy W.S.</td>
<td>0.304</td>
<td>0.510</td>
<td><strong>0.527</strong></td>
</tr>
<tr>
<td>NAACP Speech</td>
<td>0.291</td>
<td>0.503</td>
<td><strong>0.522</strong></td>
</tr>
<tr>
<td>Average</td>
<td>0.325</td>
<td>0.499</td>
<td><strong>0.562</strong></td>
</tr>
</tbody>
</table>

hard and don’t let anything stop you. Happy in Chick-fil-A” was labeled as 0. A total of 16 human evaluators were invited for answer labeling. Each topic was labeled by two evaluators, and the average score between them for each question and video pair is used as the ground-truth. On average, each subject labeled 500 question-video pairs pooled from five approaches (described in Section 4.4.3). In the experiment, we combined the human labels to simulate the ideal DCG (IDCG) and the average value of the IDCG is 28.1.

4.4.2 Opinion Localization

This section evaluates the accuracy of locating opinion clips. Table 4.2 first shows the results of detecting sentiment talking tracks using different combinations of ASR, caption (C) and metadata (M). As presented in Section 4.1.2, the prior probability for Equation 4.2 is directly derived from metadata. For the combinations without metadata, we set the prior as 0.5 in the experiment. The result shows that, by ASR-only the accuracy of detection is around 0.65. This result is boosted by 4% and 7% respectively when fusing with caption and metadata. The recognition rate of OCR is around 85%, which is the reason that caption
introduces less improvement than metadata. The best performance is attained when all the modalities are considered. From our result analysis, both caption and metadata are good supplement of ASR. For example, the relevance score for a clip entitled “We, young Americans, pay for the services and we don’t use” is weak in sentiment. But when combining with captions such as “opposer of Obamacare” extracted from the video track or the description “Why a majority of Americans remain opposed to Obamacare” from metadata, the speech potentially expresses an opinion and can be used for sentiment-based question-answering.

Table 4.3 further shows the result of opinion clip localization. We compare our approach (EM) to two baselines implemented based on k-means and random guess (Random) respectively. Both EM and k-means outperforms “Random” by a large margin. By modeling distributions of multiple feature types, EM is more probabilistically sound compared to k-means, which is biased towards generating two equi-sized clusters. As indicated in Table 4.3, EM achieves the best performance of 0.562, with an improvement of 13% than that of k-means. In addition, we also experimented the difference between EM and supervised learning algorithm. Using 20% sentiment tracks randomly selected from the eight topics as training samples, a SVM classifier with $\chi^2$ RBF kernel is learnt for classification. We compared the performance of EM and SVM on the remaining 80% of sentiment tracks. The accuracy of SVM is 0.573, which is slightly better than EM with accuracy of 0.565.

### 4.4.3 Opinion Question Answering

We compare our proposed model DeepHPam with four other methods TF-IDF, BM25 [92], partial least square (PLS) [93, 94] and DeepMatch [52]. The first three methods can be considered as linear models, while DeepHPam and DeepMatch are nonlinear models. For BM25, we set $k_1 = 1.2$ and $b = 0.75$ according to the
safe range suggested in [95]. PLS projects the TF-IDF vectors from question and answer domains into a latent subspace by linear mapping, and then measures the matching between them by dot product. Using the QA corpus mentioned in Section 4.4.1.2, PLS learns a subspace of 300 dimensions for projection. The architecture of DeepMatch is similar to DeepHPam: 999-300-20 neurons in the first, second and committee layers respectively, all with sigmoid functions. In DeepMatch, Latent Dirichlet allocation (LDA) is applied for generating the nodes (or events) at p-layerI and p-layerII separately. The connections between the nodes of two layers are adhocly determined based on word overlapping between two nodes. The parameters between p-layerI and p-layerII are randomly initialized. Note that, although DeepHPam and DeepMatch share the same representation for an input QA pair, the binary vectors (as described in the first paragraph of Section 5.2) correspond to an input are different as the underlying event models are generated
The experiment was conducted by matching text questions to the opinion clips mined in Section 4.4.2. Specifically, the ASR of a clip is extracted and then the similarity to a given question is measured. Figure 4.5 shows the result of performance comparison in terms of nDCG@10. Basically, non-linear matching of QA pairs shows an relative improvement of 56% (by DeepHPam) and 32% (by DeepMatch) over the baseline TF-IDF. Because of the lexical gap between questions and answers, only few words co-occur between them. As a result, short clips are preferred in retrieval based on TF-IDF and BM25. Obviously, short clips may not be informative enough for answering opinion questions. For example, the speech “Who cares about Romney’s tax returns? We know he is rich, that’s enough.” is retrieved as the top-1 opinion clip by BM25 for the question “Why is Mitt Romney hiding his tax returns?” While by DeepHPam, we are able to retrieve the clip “not only he hasn’t paid about thirteen percent in taxes in the years that we know, he pumping up and up in just one year all the way up to the fourteen percent now. But is there money in the cayman islands? Is there money in bermuda area, in the Swiss bank, in China and every?...” In this example, the words “tax” and “return” in the question domain are modeled together with “cayman”, “bermuda” and “bank” in the answer domain by a neuron, and thus a better answer can be successfully retrieved. Another observation is that PLS is incapable of dealing with words of complex relationship using linear projection. For example, the word “abuse” is used to describe the usage of “chemical weapons” and the “power of president to launch military strike”. This relationship can be modeled by DeepMatch and DeepHPam but not PLS.

To show the advantage of using hPam [90] for building and initializing the parameters in DeepHPam, Figure 4.6 compares the learning efficiency of DeepHPam and DeepMatch. As shown in the figure, the initial loss of DeepHPam is
Figure 4.6: Trends of ranking-based loss of DeepHPam and DeepMatch on the topics “Obamacare” and “Snowden”. The loss is summed over the entire training set. The learning rate is 0.01 and the coefficient of L2 regularization term is 0.0001. The training losses of other topics follow the similar trends.

much less and the convergence speed is quicker than DeepMatch. In addition to learning efficiency, DeepHPam consistently outperforms DeepMatch across all the topics. We browsed through and compared the opinion clips retrieved by both approaches. Our observation is that DeepHPam is more capable of retrieving video answers, which are labeled as “2” (i.e., fully answer a question), while DeepMatch is more susceptible to frequent words, which lack specificity in question answering. We attribute this to the use of hPam in constructing a more precise neuron connectivity in the deep architecture, versus DeepMatch where the connections are merely established upon overlapping words between player-I and player-II. Figure 4.7 shows examples of the opinion clips retrieved by DeepHPam. The examples range from very specific questions, such as “Do U.S. doctors like Obamacare?”, to general questions, such as “Do you think Obama will attack Syria?”

4.4.4 User Study

We are interested to know whether the retrieved video answers are more preferable than text answers. To verify this, we conducted a subjective test by providing users
<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do U.S. doctors like Obamacare? If yes or no, why?</td>
<td>My argument is it is a lie. They have lied to you before like ... you’re not gonna keep your doctor, you’re not gonna keep your insurance ... every doctor knows that this is bad, this is not gonna work because there is not enough resources</td>
</tr>
<tr>
<td>Do you think Obama will attack Syria?</td>
<td>He played a major role. He was prepared to go ... he was blindside by that decision of British parliament and then to find out that NATO said no ... push him to recognize he could not just go ahead with military strike ...</td>
</tr>
<tr>
<td>Americans do you consider Edward Snowden as Hero?</td>
<td>Edward Snowden, I think he is a hero. Cuz he expose the corruption and this administration once a triumph for treason ... as being as corrupt as they are .. Be exempt by American ...</td>
</tr>
<tr>
<td>How will the government shutdown affect us?</td>
<td>They won’t effect me personally ... but I know I have several friends that are children of military families ... pay for school ... they have to stop and feel very devastating ...</td>
</tr>
</tbody>
</table>

Figure 4.7: Example of opinion clips retrieved by DeepHPam. The keyframes and partial transcripts of the opinion clips are shown for illustration purpose.

Both video and text answers, and asking them to pick the preferred answer medium. A total of 12 subjects (5 females and 7 males) from different education background, including computer science (7), biology (1), chemistry (1), and business (3), were invited for the user study. The average age of the subjects is 28, and the ages range from 25 to 32. Each subject was assigned 8 questions and 16 answers. Only questions that have the best text answers picked by the askers were selected for testing. During the test, an subject was shown with a question, along with the best text answer and the top-1 video answer retrieved by our DeepHPam. Besides picking a preferred answer, an subject was also asked to provide a score, in the range of 1 to 3, judging how good the preferred answer is better than the other one. The score “2” (“3”) means a (definitely) better answer, and “1” means both answers are indeed comparable and not significantly differ. The scores between “2” and “3” differ in the degree of preference, while “1” could indicate a selection
The result of user studies shows that 56.70% of selections indicate that video answer is a better medium, with an average score of 1.82 for the questions where video answers were picked. For text answers, the average score is 1.32. Among all the 96 selections, there are 22.68% (6.19%) of times where video (text) answers were picked with a score of 3. The result basically indicates that video answers are preferred, but not exceeding the threshold of better or definitely better category, in most of the cases.

We further conducted ANalysis Of VAriance (ANOVA) test to evaluate the level of significance. The first evaluation is about the significance of result, i.e., the hypothesis that there is no difference between text and video answers. By F-test statistics, the value is 21.64 at the significance level of \( p = 6.13 \times 10^{-6} \). In other words, the hypothesis is rejected, indicating that video answers are significantly better than text answers in the user study. The second evaluation tests the consistency in user rating, with the hypothesis that the variance among the ratings of users is not significant. The value of F-test statistic is 0.35 with significance level of \( p = 0.95 > 0.1 \), showing the difference among users is statistically insignificant.

The statistics from ANOVA basically supports our claim that video answers are likely to be preferred for opinion-oriented questions. Here we show one interesting example receiving high score in the study. Referring to the question on the top-left of Figure 4.7, the best text answer is “Of course not, that’s why so many will retire early. They know more about it than the lib basement occupiers here.” The video answer, which shows an opinion holder speaking in a dramatic tone while arguing the question from the perspective of doctor, receives a score of 3 compared to the text answer. We observed that most video answers received high scores are able
to provide additional information not in text answers, in addition to vivid display of emotion in expressing opinion.

4.5 Summary

We have presented our attempt in searching sentiment-oriented clips for answering opinion questions of controversial topics. Two major difficulties in building such a system, i.e., answer localization and QA matching, are respectively tackled with the multi-modal analysis of emotion content with heuristics and the nonlinear matching of speech and text with deep architecture. In general, finding a right segment from a video collection as an answer (analogy to finding a sentence from a document collection) is a highly difficult problem, particularly when the questions of interest are about opinion expression. Our proposed solution, with the strategy of trimming the search space by considering only sentiment tracks with opinion holders as candidates for matching, is shown to work effectively and fits well for questions “born with emotions” – which there are many such examples of questions, as posted on the social media platforms such as Yahoo! Answers. The employment of deep learning in matching text questions with video answers is also a new attempt in the literature, and we are able to demonstrate encouraging results leveraging on the power of nonlinear and hierarchical matching.
CHAPTER 5

DEEP MULTIMODAL LEARNING FOR AFFECTIVE ANALYSIS AND RETRIEVAL

Social media is an opinion-rich knowledge source including plenty of timely user-generated-contents (UGC) with different media types. Automatically understanding the emotional status of users from their uploaded multimedia contents is in high demands for many applications [5]. For example, when searching information about a resort, the retrieved images or videos can be ranked based on their emotions to provide implicit comments. In addition, when asking opinion-related questions about hot events, providing emotion tags for retrieved videos helps users more quickly understand the sentiment of public’s view. This function can also be used by governments to better understand people’s reactions towards their new policies.

The existing works on affective analysis of UGC data are mostly devoted to single media [53, 54, 55, 56, 1, 57, 58, 59, 60]. For example, linguistic [53, 55] and semantic [54, 56] features are both adopted for text-based analysis. However, inferring perceived emotion signals underlying short messages that are usually sparse in textual description is not easy. Figure 5.1 (1st row) shows two images of “sadness” and “happiness” emotions respectively. Nevertheless, no obvious emotion clues are observed by merely reading their text captions, which play the roles of referring visual content (roller coaster) rather than emotion in the images. In other words, the captions convey semantic meanings while the actual emotion signals are buried inside the images. Apparently, visual content such as color contrast and tone provide more vivid clues to reveal the underlying emotions for
Figure 5.1: Examples of emotional images and videos with their associated tags and titles from Twitter and YouTube. The left image in the first row describes the destruction caused by Hurricane Sandy and the right one describes the Manta roller coaster in the seaworld Orlando. The textual descriptions for these two images are almost the same and the implicit emotions can only be predicted by the visual information. In contrast, emotion classification in the lower two videos with similar visual appearances is performed using the strong clues in the textual captions and auditory information.

Due to these limitations, there are several works studying the fusion of multi-modal features, including multi-kernel fusion [5], conditional random field [67] and Bayesian rules [66]. These works are based on the standard early or late fusion strategies [65, 66, 67, 64], despite the employment of different machine learning models. The “shallow” way of combining different features is also questionable
in principle, given the diverse representations and correlational structures among different modalities. For example, it remains an open problem on the right way of combining raw pixels and audio waveform, for joint understanding of phonemes and visemes (lip pose and motion), in speech recognition [96]. In brief, the highly non-linear relationships existing between different modalities, particularly, are often overlooked by the existing approaches.

Emotion is also correlated with surrounding context, specifically the objects, sounds and scenes in images or videos. For example, a video with “a man listening to bird chirping in the garden” casts an enjoy mood useful for emotion prediction. SentiBank [1] is one such recent effort that explicitly identifies 3,244 adjective noun pairs (ANPs) for learning large-scale emotion-oriented concept classifiers. Examples of ANPs are “amazing flowers”, “awesome view” and “shy smile”. Nevertheless, due to the open nature of how nouns and adjectives are combined, SentiBank is hard to be generalized to cover the possible ANPs, not even mentioning the daunting efforts required in labeling of training examples for training ANP classifiers free of sample noise.

In this chapter, we propose a more generalized framework for unsupervised feature learning based on Deep Boltzmann machine (DBM) [97], aiming to learn features coupling emotion and semantic signals buried in multimodal signals. As we consider eight types of wildly different features in terms of statistical properties, deep-based learning is preferable for inferring non-linear relationships among features within and across modalities. A joint embedded space shared by multimodal signals is expected to capture such relationships with semantic and emotion contexts. As studied in other works [98], a joint space learnt by DBM is capable of preserving both common and modality-specific information. The learning of DBM is unsupervised and thus is suitable for our problem as plenty of weakly labeled training examples are freely available on social media websites. Although these
Figure 5.2: The multimodal DBM that models the joint distribution over visual, auditory and textual features. All layers but the first (bottom) layers use standard binary units. Gaussian RBM model is used to model the distributions over the visual and auditory features. Replicated Softmax topic model is applied on the textual features.

Examples are without careful hand-labels, the text captions can still somewhat provide rich learning signals for mapping diverse visual and auditory features to a coherent embedded space shared by different modalities. For example, the text captions in Figure 5.1 provide clue connecting two visually dissimilar roller coasters. The word “joy” can possibly link diverse events, such as wedding party and couple hugging, of different audio-visual effects.

Traditionally emotion prediction and semantic classification are treated as two separate tasks. Emotion-oriented queries such as “dog hates bath” always posts challenges for no clear understanding of how classifiers of different natures should be combined. SentiBank[1] could possibly deal with such queries, but is inherently limited by the number of ANP vocabularies. Our model, empowered on the joint space learnt through DBM, can more naturally answer these queries, assuming the availability of a huge number of training examples with wild but rich textual, visual and auditory signals. Due to unsupervised learning, our model has a much better capacity and scalability than SentiBank in capturing emotional experience.
in multimodal settings. Furthermore, the joint space also enlightens cross-modal retrieval, where for example, either text-to-video or video-to-text search can be performed under our model.

The main contribution of this work is the proposal of a deep multimodal learning platform that enables a more generalized way of learning features coupled with emotions and semantics. Empirical studies also demonstrate the feasibility of employing such features for multi-modal affective classification and retrieval. The remaining of this chapter is organized as follows. Section 5.1 presents the deep network architecture, followed by sections 5.2 and 5.3 describing the joint space representation and network learning respectively. Section 5.4 and 5.5 presents experimental results on affective analysis and retrieval respectively, and finally Section 5.6 concludes this chapter.

5.1 Deep Network Design

Figure 5.2 shows the proposed network architecture, which is composed of three different pathways respectively for visual, auditory and textual modalities. Each pathway is formed by stacking multiple Restricted Boltzmann Machines (RBM), aiming to learn several layers of increasingly complex representations of individual modality. Similar to [2], we adopt Deep Boltzmann Machine (DBM) [97] in our multimodal learning framework. Different from other deep networks for extracting feature, such as Deep Belief Networks (DBN) [99] and denoising Autoencoders (dA) [100], DBM is a fully generative model which can be utilized for extracting features from data with certain missing modalities. Additionally, besides the bottom-up information propagation in DBN and dA, a top-down feedback is also incorporated in DBM, which makes the DBM more stable on missing or noisy inputs such as weakly labelled data on the Web. The pathways eventually meet
and the sophisticated non-linear relationships among three modalities are jointly learned. The final joint representation can be viewed as a shared embedded space, where the features with very different statistical properties from different modalities can be represented in an unified way.

The proposed architecture is more generalized and powerful in terms of scale and learning capacity. In visual pathway, the low-level features amount to 20,651 dimensions, resulting in large number of parameters to be trained if connecting them directly to the hidden layer. Instead, we design a separate pathway for each low-level feature, which requires less parameters and hence more flexible and efficient to train. This advantage makes our system more scalable to handling higher dimensional features, rather than features of 3,875 dimensions used in [2].

We further consider learning the separated pathways in visual modality in parallel. The computational cost can be further reduced. Furthermore, we generate a compact representation which represents the common feature and preserves the unique characteristic of each visual feature. In this way, it will not overwhelm other modalities because of high dimensionality during joint representation learning. Auditory and textual pathway do not suffer from this problem. However, the proposed structure can be easily extended for other modalities.

It is worth noticing that the high-level semantics in visual and auditory modalities can be represented in the final joint representation, by considering the correlations between them and textual inputs during training. Since our model is fully generative, the semantics of input data without textual modality can also be extracted. Other semantic features for visual and auditory data (e.g., SentiBank [1], Classemes [101] and Objectbank [102]) basically adopt the shallow learning models, which learn the local patterns extracted from the data. These methods suffer from information loss [103, 104], and is sensitive to the diverse appearance of input data. In contrast, our model has the capability of mining generative repre-
sentations from the raw data, which has been proved to be more powerful [105, 98].

5.2 Multimodal Joint Representation

Our network is built upon RBMs. A standard RBM has two binary-valued layers, i.e., visible layer (denoted as $v$) and hidden layer (denoted as $h$). The probability distribution over the inputs $v$ is defined as

$$P(v) = \frac{1}{Z} \sum_h \exp(-E(v, h)).$$

(5.1)

$E(v, h)$ is the free energy between $v$ and $h$, given by

$$E(v, h) = -\sum_{ij} v_i w_{ij} h_j - \sum_i a_i v_i - \sum_j b_j h_j,$$

(5.2)

where $w_{ij}$ is the weight of link connecting two layers, $a$ and $b$ are the bias weights for $v, h$ respectively. The feature learning problem is elegantly stated to maximize the probability in Equation 5.1 or to minimize the free energy in Equation 5.2. The standard RBM can only handle binary-valued inputs. Other generalized RBMs include Gaussian RBM [106] designed for modeling real-valued inputs and Replicated Softmax [107] for modeling sparse word count vectors. Next, we describe different pathways and their joint representation. Each pathway consists of a stack of RBMs selected according to the property of input data.

5.2.1 Visual Pathway

The visual input consists of five complementary low-level features widely used in previous works [5, 1]. As shown in Figure 5.2, each feature is modeled with a separate two-layer DBM. Let $\mathcal{K} = \{d, g, o, l, s\}$ denote the set of five features, respectively as DenseSIFT [108], GIST [109], HOG [110], LBP [111] and SSIM
Furthermore, let \( v_\psi = \{v^k\}, h_\psi^1 = \{h^{(1k)}\} \) and \( h_\psi^2 = \{h^{(2k)}\} \) as the sets of real-valued inputs, first and second hidden layers respectively, where \( k \in \mathcal{K} \). For example, \( v^d \) refers to the visible layer for DenseSIFT. In addition, the joint layer in visual pathway (the layer in red in Figure 5.2 is denoted as \( h^{(v)} \)).

The connections between \( v^k \) and \( h^{(1k)} \) are modeled with Gaussian RBM \([106]\) and the connections between \( h^{(1k)} \) and \( h^{(2k)} \) are modeled with standard binary RBM. Hence, the probability distribution over the real-valued input \( v^k \) is given by

\[
P(v^k; \theta^k) = \frac{1}{\mathcal{Z}(\theta^k)} \sum_{h^{(1k)}, h^{(2k)}} \exp(-E(v^k, h^{(1k)}, h^{(2k)}; \theta^k)) \tag{5.3}
\]

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

\[
E(v^k, h^{(1k)}, h^{(2k)}; \theta^k) = \sum_i \frac{(v^k_i - b^k_i)^2}{2(\delta^k_i)^2} - \sum_{ij} \frac{v^k_i}{\delta^k_i} W_{ij}^{(1k)} h_j^{(1k)} - \sum_{jl} h_j^{(1k)} W_{jl}^{(2k)} h_j^{(2k)} 
\tag{5.4}
\]

where \( \theta^k = \{a, b, W^{(1k)}, W^{(2k)}\} \) are the model parameters. Note that for brevity, the bias terms \( a \) on the hidden layers are omitted. To generate the joint representation over these five low-level features, we combine the five DBM models by adding an additional layer \( h^{(v)} \) on top of them. Then, the joint density distribution over the five features \( v_\psi \) is given by

\[
P(v_\psi; \theta^{(v)}) = \sum_{h_\psi^2, h^{(v)}} P(h_\psi^2, h^{(v)}) \left( \prod_{k \in \mathcal{K}} \left( \sum_{h^{(1k)}} P(v^k, h^{(1k)} | h^{(2k)}) \right) \right) 
\tag{5.5}
\]

The density distribution \( P(h_\psi^2, h^{(v)}) \) in Equation 5.5 is given by

\[
P(h_\psi^2, h^{(v)}) = \frac{1}{\mathcal{Z}(\theta^{(v)})} \prod_{k \in \mathcal{K}} \exp \left( \sum_{pq} W^{(3k)} h_p^{(2k)} h_q^{(v)} \right) 
\tag{5.6}
\]

The joint distribution \( P(v^k, h^{(1k)} | h^{(2k)}) \) of \( v^k \) and \( h^{(1k)} \) over \( h^{(2k)} \) in Equation 5.5
can be easily inferred from Equation 5.3 as

\[
P(v^k, h^{(1k)} \mid h^{(2k)}) = \frac{1}{Z(\theta^k)} \exp\left(- \sum_i \frac{(v^k_i - b^k_i)^2}{2\delta_i^2}\right) + \sum_{ij} v^k_i W_{ij} h^{(1k)}_j + \sum_{jp} W_{jp}^{(2k)} h^{(1k)}_j h^{(2k)}_p
\]  

(5.7)

Until now, all the probability distributions in Equation 5.5 are provided and the probability distribution over the whole set of input features \(v_V\) in visual pathway can be easily inferred by subscribing these equations.

5.2.2 Auditory Pathway

The input features adopted in auditory pathway are MFCC \([113]\) and Audio-Six (i.e., Energy Entropy, Signal Energy, Zero Crossing Rate, Spectral Rolloff, Spectral Centroid, and Spectral Flux) \([5]\). The Audio-Six descriptor, which can capture different aspects of an audio signal, is expected to be complementary to the MFCC. Since the dimension of Audio-Six is only six, we directly concatenate the MFCC feature with Audio-Six rather than separating them into two sub-pathways as the design in visual pathway. The correlation between these two features can be learned by the deep architecture of DBM \([2]\). Let \(v_a\) denote the real-valued auditory features and \(h^{(1a)}\) and \(h^{(2a)}\) represent the first and second hidden layers respectively. Similar to Equation 5.3, the DBM is constructed by stacking one Gaussian RBM and one standard binary RBM.

5.2.3 Textual Pathway

Different from the visual and auditory modalities, the inputs of the textual pathway are discrete values (i.e., count of words). Thus, we use Replicated Softmax \([107]\) to model the distribution over the word count vectors. Let \(v_t\) as a visible unit denoting the associated metadata (i.e., title and description) of a video \(t\), and \(v^k_t\) denotes the count of the \(k^{th}\) word in a pre-defined dictionary containing \(K\) words. The first and second hidden layers are \(h^{(1t)}\) and \(h^{(2t)}\). Then, the probability of
generating $\mathbf{v}_t$ by the text-specific two-layer DBM is given by

$$
P(\mathbf{v}_t; \theta^t) = \frac{1}{Z(\theta^t)} \sum_{\mathbf{h}^{(1t)}, \mathbf{h}^{(2t)}} \exp \left( \sum_{jk} W_{kj}^{(1t)} h_j^{(1t)} v^t_k \right) + \sum_{jl} W_{jl}^{(2t)} h_j^{(1t)} h_l^{(2t)} + N \sum_j b_j^{(1t)} h_j^{(1t)} \right) \tag{5.8}
$$

Note that the bias term of the first hidden layer $\mathbf{h}^{(1t)}$ is scaled up by the length of the document. This scaling is important for allowing hidden units to behave sensibly when dealing with documents of different lengths. As stated in [2, 107], without the bias scaling, the scale of the weights would be optimized to fit to the average document length. This would induce that the longer documents tend to saturate the units and shorter ones may be ambiguous on activating the hidden units.

### 5.2.4 Joint Representation

To combine the learned representations of DBMs for the three modalities, an additional layer is added on top of the three pathways, which is annotated as “Joint Representation” in Figure 5.2. We denote this layer as $\mathbf{h}^{(J)}$. We further use $\mathbf{v} = \{\mathbf{v}_V, \mathbf{v}_a, \mathbf{v}_t\}$ to represent all the visible inputs. The final joint density distribution over multi-model inputs can be written as

$$
P(\mathbf{v}; \theta) = \sum_{\mathbf{h}^{(v)}, \mathbf{h}^{(2a)}, \mathbf{h}^{(2t)}, \mathbf{h}^{(J)}} P(\mathbf{v}^{(V)}, \mathbf{h}^{(2a)}, \mathbf{h}^{(2t)}, \mathbf{h}^{(J)})
$$

$$
(\sum_{\mathbf{h}^{(v)}, \mathbf{h}^{(2a)}, \mathbf{h}^{(2t)}} P(\mathbf{v}^{(V)}, \mathbf{h}^{(2a)}, \mathbf{h}^{(2t)} | \mathbf{h}^{(v)}))(\sum_{\mathbf{h}^{(1a)}} P(\mathbf{v}_a, \mathbf{h}^{(1a)} | \mathbf{h}^{(2a)}))
$$

$$
(\sum_{\mathbf{h}^{(1t)}} P(\mathbf{v}_t, \mathbf{h}^{(1t)} | \mathbf{h}^{(2t)}))
$$

By subscribing equations 5.5 and 5.8 into above equation, the probability distribution over the multiple inputs formulated by the proposed network can be easily inferred.
5.3 Networking Learning and Inferencing

5.3.1 Approximate Network Learning

The learning of our proposed model is not trivial due to multiple layers of hidden units and multiple modalities. Inspired by [2], we split the learning process into two stages. First, each RBM component of the proposed multimodal DBM is pretrained by using the greedy layerwise pretraining strategy [97]. In this stage, the time cost for exactly computing the derivatives of the probability distributions with respect to parameters increases exponentially with the number of units in the network. Thus, we adopt 1-step contrastive divergence (CD$_1$), an approximate learning method, to learn the parameters. In CD$_k$ algorithm, a $k$-step Markov chain is initialized with the training sample. The stochastic reconstruction of the training sample from Markov chain by Gibbs sampling has a decreased free energy. Hence, this reconstruction can be approximately treated as the distribution generated by the RBM model. The contrast between the training sample and its reconstruction is used to approximate the direction of the change for the parameters. In practice, CD$_1$ is widely used for RBM training, since good approximation of the changing direction is already obtained when $k = 1$. Note that, CD$_1$ actually performs poorly in approximating the size of the change in parameters. However, it is accurate enough for learning a RBM to provide hidden features for a high-level RBM training [114]. This is because CD$_1$ retains most of the information in the inputs.

As discussed in [114], CD$_1$ is still far from optimal to be used for learning a joint-density model. Therefore, in the joint learning stage, we adopt a more radical departure from CD$_1$, named as “persistent contrastive divergence” (PCD) [115]. In contrast to initialize each alternating Gibbs Markov chain at a training sample, the states of a number of persistent chains or “fantasy particles” are tracked in PCD. Each persistent chain has its hidden and visible states, which are generated by running mean-field updates with Gibbs sampling for one or a few times after each weight is updated. Then the derivative of the probability distribution is approximated by the difference between the pairwise statistics measured on a mini-batch of data and the persistent chains. Since the weight-updates repel each chain from its current state by raising the energy of that state, the persistent chains mix surprisingly fast [116]. Furthermore, PCD also learns significantly better models than CD$_1$ or even CD$_{10}$ as reported in [115].
5.3.2 Joint Representation Inferring

The representation learnt by the proposed model is a set of distributions over layers conditioned on their adjacent layers. Based on the Equations 5.7 and 5.8. The conditional distributions over all the layers along the “DenseSIFT” and “Textual” pathway is inferred as following. The inference on other pathways follows the similar process. For compactness, the bias term for each layer is omitted in Equation 5.10.

\[ v_i^d | h^{(1d)} \sim N(\delta_i \sum_j W_{ij}^{(1d)} h_j^{(1d)} + b_i^d, \delta_i^2) \]

\[ p(h_j^{(1d)} = 1 | v^d, h^{(2d)}) = g(\sum_i W_{ij}^{(1d)} v_i^d + \sum_l W_{jl}^{(2d)} h_l^{(2d)}) \]

\[ p(h_l^{(2d)} = 1 | h^{(1d)}, h^{(v)}) = g(\sum_j W_{jl}^{(2d)} h_j^{(1d)} + \sum_q W_{lq}^{(v)h^{(v)}} h_q^{(2g)}) + \sum_l W_{lq}^{(v)h^{(v)}} h_l^{(2g)} + \sum_l W_{lq}^{(v)h^{(v)}} h_l^{(2s)}) \]

\[ p(v_{ik} = 1 | h^{(1t)}) = \frac{\exp(\sum_j h_j^{(1t)} W_{jk}^{(1t)} + b_k^t)}{\sum_{q=1}^K \exp(\sum_j h_j^{(1t)} W_{jq}^{(1t)} + b_k^t)} \] (5.10)

\[ p(h_j^{(1t)} = 1 | v^t, h^{(2t)}) = g(\sum_{k=1}^K W_{kq}^{(1t)} v_k^t + \sum_l W_{jl}^{(2t)} h_l^{(2t)} + Nb_j^{(1t)}) \]

\[ p(h_l^{(2t)} = 1 | h^{(1t)}, h^{(J)}) = g(\sum_j W_{jl}^{(2t)} h_j^{(1t)} + \sum_p W_{lp}^{(Jt)} h_p^{(J)}) \]

\[ p(h_p^{(J)} = 1 | h^{(v)}, h^{(2a)}, h^{(2t)}) = g(\sum_q W_{qp}^{(Jv)} h_q^{(v)} + \sum_l W_{lp}^{(Ja)} h_l^{(2a)} + \sum_m W_{mp}^{(Jt)} h_m^{(2t)}) \]

where \( g(x) = 1/(1 + \exp(-x)) \) is the logistic function. When inferring the distributions, the observed modalities are clamped at the inputs and Gibbs sampling is performed for updating the states of each layer. As mentioned in Section 5.3.1,
Table 5.1: The number of neurons in each layer of our enhanced multimodal DBM (E-MDBM). $v, h^1, h^2, h^{(v)}$ and $h^{(J)}$ represent the visible layers, first hidden layers, second hidden layers, joint representation layer over visual features and joint representation layer over visual, auditory and textual modalities.

<table>
<thead>
<tr>
<th>Features</th>
<th>$v$</th>
<th>$h^1$</th>
<th>$h^2$</th>
<th>$h^{(v)}$</th>
<th>$h^{(J)}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DenseSIFT</td>
<td>6,300</td>
<td>2,048</td>
<td>1,024</td>
<td>2,048</td>
<td>4,096</td>
</tr>
<tr>
<td>GIST</td>
<td>512</td>
<td>1,024</td>
<td>1,024</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HOG2x2</td>
<td>6,300</td>
<td>2,048</td>
<td>1,024</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LBP</td>
<td>1,239</td>
<td>2,048</td>
<td>1,024</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSIM</td>
<td>6,300</td>
<td>2,048</td>
<td>1,024</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MFCC+AudioSix</td>
<td>4,000</td>
<td>2,048</td>
<td>1,024</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Word Count Vector</td>
<td>4,314</td>
<td>2,048</td>
<td>1,024</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Mean-field update is adopted for state updating. Since each hidden layer is influenced by its higher and lower layers, alternating sampling is conducted for update all the necessary states to approximate the distribution. If all the modalities are present, the joint representation can be easily inferred by Gibbs sampling from $p(h^{(J)} = 1 | h^{(v)}, h^{(2a)}, h^{(2t)})$.

There are two ways to generate the joint representation if some modalities are not available. First we can directly generate the joint representation based on the existing modalities only and leave the missing ones out. For example, if text is missing, the joint representation will be computed by $p(h^{(J)} = 1 | h^{(v)}, h^{(2a)})$. The second way is to infer the missing modalities by alternating Gibbs sampling. Meanwhile, the joint representation is updated with the generated data of missing modalities. For example, assuming that the textual modality is missing, the observed visual modality $v_v$ and auditory modality $v_a$ are clamped at the inputs and all hidden units are initialized randomly. Alternating Gibbs sampling is used to draw samples from $P(v_t | v_v, v_a)$ by updating each hidden layer given the states of the adjacent layers. As reported in [2], the second method achieves better performance than the first one, which indicates that the multimodal DBM can generate meaningful representations of the missing modalities.
5.3.3 Discussions

While the proposed architecture follows the principle [2], the main novelty comes from the design of multiple visual pathways. Despite that the architecture may appear more complicated than [2] at first glance, the design indeed simplifies [2] by significantly reducing the number of hyper parameters. With reference to Table 5.1 that lists the number of neurons in each layer, the network contains 99,690,496 hyper parameters. While this number is terribly high, it requires only 29% of that parameters in [2], for converting the input features from 20,651 to 2,048 dimensions. Furthermore, the design accelerates learning by allowing parallel training of parameters on each pathway. The locally connected hidden units in the pathways also speed up the PCD learning at the second stage. In our experiments, by using Tesla-K20 GPU, network learning completes in about one week with around 1 million training examples. Given the same amount of time, [2] is only able to train a network with input features of 3,875 dimensions. In short, our proposed network is more scalable in learning and effective in testing (see Section 5.4) than [2].

Overfitting becomes an issue with large number of hyper parameters to be learnt in the network. As stated in [114], assuming that each image contains 1,000 pixels, using 10,000 training examples to learn weights of a million parameters in one RBM is quite reasonable. In the proposed network, the largest RBM has $6,300 \times 2,048$ or around 1.3 millions of parameters. Using 20,000 training samples, for example, is practically feasible for learning this RBM. Since RBMs in the network are learnt in parallel, the chance of overfitting shall not be high even with only around 20,000 training samples. In our case, there are close to a million of training images and videos (see Section 5.4.1), and we did not observe tendency of overfitting when learning the parameters.

5.4 Experiment: Affective Analysis

This section starts by introducing model training with unlabeled images and videos sampled from social media websites (Section 5.4.1). Two sets of experiments are conducted for affective analysis (Section 5.4.2 and Section 5.4.3), respectively emotion prediction on YouTube videos and sentiment classification on Twitter images.
Table 5.2: Prediction accuracies for each emotion category of VideoEmotion obtained by applying logistic regression to representations learned at different hidden layers. The highest accuracy of each category is highlighted.

<table>
<thead>
<tr>
<th>Category</th>
<th>(h^{(1d)})</th>
<th>(h^{(2d)})</th>
<th>(h^{(1g)})</th>
<th>(h^{(2g)})</th>
<th>(h^{(1o)})</th>
<th>(h^{(2o)})</th>
<th>(h^{(1l)})</th>
<th>(h^{(2l)})</th>
<th>(h^{(1s)})</th>
<th>(h^{(2s)})</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>0.152</td>
<td>0.242</td>
<td>0.061</td>
<td>0.152</td>
<td>0.273</td>
<td>0.182</td>
<td>0.273</td>
<td>0.333</td>
<td>0.242</td>
<td>0.212</td>
</tr>
<tr>
<td>Anticipation</td>
<td>0.031</td>
<td>0.094</td>
<td>0.031</td>
<td>0.031</td>
<td>0.125</td>
<td>0.062</td>
<td>0.062</td>
<td>0.031</td>
<td>0.031</td>
<td>0.031</td>
</tr>
<tr>
<td>Fear</td>
<td>0.436</td>
<td>0.418</td>
<td>0.455</td>
<td>0.445</td>
<td>0.345</td>
<td>0.309</td>
<td>0.418</td>
<td>0.509</td>
<td>0.145</td>
<td>0.345</td>
</tr>
<tr>
<td>Joy</td>
<td>0.373</td>
<td>0.441</td>
<td>0.339</td>
<td>0.441</td>
<td>0.390</td>
<td>0.373</td>
<td>0.356</td>
<td>0.373</td>
<td>0.407</td>
<td>0.339</td>
</tr>
<tr>
<td>Sadness</td>
<td>0.176</td>
<td>0.235</td>
<td>0.029</td>
<td>0.118</td>
<td>0.059</td>
<td>0.059</td>
<td>0.059</td>
<td>0.206</td>
<td>0.206</td>
<td>0.265</td>
</tr>
<tr>
<td>Surprise</td>
<td>0.675</td>
<td>0.650</td>
<td>\textbf{0.863}</td>
<td>0.775</td>
<td>0.750</td>
<td>0.713</td>
<td>0.762</td>
<td>0.700</td>
<td>0.750</td>
<td>0.850</td>
</tr>
<tr>
<td>Trust</td>
<td>0.062</td>
<td>0.156</td>
<td>0.031</td>
<td>0.031</td>
<td>0.062</td>
<td>0.031</td>
<td>0.125</td>
<td>0.094</td>
<td>\textbf{0.188}</td>
<td>0.094</td>
</tr>
<tr>
<td>Overall</td>
<td>0.327</td>
<td>0.365</td>
<td>0.313</td>
<td>0.313</td>
<td>0.338</td>
<td>0.343</td>
<td>0.332</td>
<td>0.346</td>
<td>0.352</td>
<td>0.360</td>
</tr>
</tbody>
</table>

5.4.1 Model Learning

We constructed two datasets, E-Flickr and E-YouTube, for DBM learning. The images in E-Flickr are crawled from Flickr by using the 3244 ANPs used in SentiBank [1] as keywords. On average, there are 250 images being retrieved for each ANP. The number of images per ANP is kept to about the same so as not to bias any ANP during DBM learning. All these images, along with their metadata (title, descriptions and tags), are included in E-Flickr. Similarly for E-YouTube, ANPs keywords are issued to YouTube for crawling videos. For each ANP, only top-100 ranked videos are considered, considering that videos further down the ranked list are likely to be irrelevant. Among these videos, lengthy videos with duration more than two minutes are excluded from E-YouTube. Generally lengthy videos are more likely to contain segments with no emotional content. Including these videos into training will practically hurt the learning effectiveness. Since tags are not available for download, each video is crawled along with title and description only. On average, 50 videos are crawled per ANP and this number does not differ by more than 10 across ANPs. To this end, E-Flickr and E-YouTube include 830,580 images and 156,219 videos respectively.

The set of features extracted from the datasets are summarized in Table 5.1. For each video, keyframes are sampled at the rate of one frame per second. Five different visual features, as listed in Table 5.1, are respectively extracted from the keyframes and then averagely pooled to form feature vectors. Audio features are extracted over every 32ms time-window of audio frames, with 50% overlap between two adjacent windows. Similar as visual features, these features are averagely pooled across time-windows. Among the set of audio-visual features, DenseSIFT,
HOG, SSIM and MFCC are further quantized into bag-of-words representation. We followed the same settings as [5], and the dimensions of different features are listed in the second column of Table 5.1. The number of neurons in the proposed architecture are designed based on the guidelines in [114]. Note that we map the GIST and LBP features into higher dimensional space so that they have more or less similar number of dimensions as other features. This is practically important to guarantee the effectiveness of learning correlations among different features. As for textual features, a total of 1,447,612 distinct words are extracted from E-Flickr and E-YouTube after stopword removal and lemmatization using CoreNLP [117]. In the experiments, only words with document frequency larger than 800 are kept. Eventually, textual feature is in 4,314 dimensions, with an average of 13 words per image and 8 words per video.

Note that video-level metadata only describes a fraction of video keyframes, and furthermore, feature pooling could possibly introduce noise. In contrast, image-level metadata provides relatively more specific description of image content and emotion. For this consideration, the learning of DBM is started from using image samples followed by video sample. Audio-pathway is left out during pre-training using E-Flickr images, but turned on when E-YouTube videos are involved for fine-tuning. During training, each dimension of visual and auditory features is mean-centered and normalized to unit variance to avoid the instability problem [114]. In addition, to avoid running separate Markov chains for each word count to get sufficient statistics for modeling distribution, all word count vectors are scaled so that they sum to 5 [2].

In this section, we evaluate the performance of the joint representation learnt using our multimodal DBM on affective analysis. We name our model as E-MDBM since the architecture has been enhanced with more features and modalities.

### 5.4.2 Video Emotion Detection

The “VideoEmotion” dataset consists of 1,101 videos which are manually labeled with eight emotional categories. Following [5], the results are evaluated by accuracy. As the textual information of the videos is not provided, we only have visual and auditory modalities in this dataset.

**Effect of exploring multimodal relations.** We first evaluate the capability of proposed model in learning non-linear relations among different modalities. The
input to the textual pathway is missing and initialized to zeros. As described in Section 5.3.2, the model is allowed to update the state of the textual input layer when performing mean-field update by alternating Gibbs sampling. In this experiment, we run the mean-field update for 5 times [2, 97]. The final joint representations (up layer) are drawn from $P(h^{(j)}|v, v_a)$ (see Equation 5.10), and used for learning a logistic regression model. For comparison, classifiers using the same training data are learned with the representations extracted from different hidden layers in Figure 5.2.

We adopt the same settings for train-test splits in [5]. Ten train-test splits are generated, each using 2/3 of the data for training and 1/3 for testing. Table 5.2 shows the prediction results. We can see that the joint representation $h^{(J)}$ achieves the best overall performance. It improves the accuracy over the joint visual representation $h^{(v)}$ and the representation from second auditory hidden layer $h^{(2a)}$ by 8.02% and 41.26% respectively. Although single modality may perform slightly better than the joint representation for some emotions, the performance is not consistent. For example, auditory feature is better to recognize “Disgust”, but it performs poorly for emotion “Fear”. This is because that “Fear” is not apparently conveyed in the auditory signal. However, visual feature achieves the best result on “Fear”. Another interesting observation is that the performance using second hidden layer of each pathway is generally better than that of the first hidden layer. As mentioned in Section 5.1, the E-MDBM model is a fully generative model. The neurons of hidden layers will receive messages from both lower layers and higher layers. By using this top-down feedback, the higher hidden layers can deal with the impact from ambiguous inputs, and thus are more robust. In addition, the joint representation ($h^{(v)}$) on visual pathway leads to 2.5% improvement over the best performance achieved in single visual feature. This indicates that the proposed structure can preserve the capability of learning correlation between different visual features, meanwhile reduce the complexity of the model learning comparing to [2].

Figure 5.3(a) further shows the confusion matrix based on the joint representation $h^{(j)}$. Most categories are confused with the category “Surprise”, where similar observation is also noted in [5]. Second, the category “anticipation” is confused particularly by “Fear” and “Surprise”. As shown in Table 5.2, almost all features perform poorly on this category. We attribute this unsatisfactory performance to
the fact that neither audio nor visual can concretely describe the emotion of “anticipation”, for example, in a sport event. Facial expression seems to be the dominant cue in conveying “anticipation”. This is probably the reason that LBP, often being applied for face recognition, performs comparatively good for this category.

Impact of missing modality. There is no textual modality in this dataset. We further evaluate the impact of missing certain modality by using either only visual feature or auditory feature as input, which are named as E-MDBM-V and E-MDBM-A respectively. The input of missing modality is initialized with zero and updated in the same way with the missing textual modality. The results are showed in Figure 5.4. We also show the result of $h^{(J)}$ in Table 5.2, which is named as E-MDBM-VA for consistency. Additionally, the performance of SentiBank from [5] and art features (AF) from [4] are also shown here for comparison. In SentiBank, logistic regression model is trained on the scores of the 1,200 ANP classifiers. For AF, 10 features are proposed in [4] for representing 6 artistic principles (i.e., balance, emphasis, harmony, variety, gradation, and movement) and logistic regression model is learnt on these features. Although our model is able to fill in the missing modalities and integrate the information into the final joint representation, E-MDBM-VA using both visual and auditory inputs exhibits the best performance. This is not surprising since the filling of missing modalities generated some noises comparing to the real data. However, the accuracy of E-MDBM-V is approaching that of E-MDBM-VA. For category “surprise”, E-MDBM-V is even better. This indicates that the missing data is somehow recovered using this model.

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Anger</th>
<th>Anticipation</th>
<th>Disgust</th>
<th>Fear</th>
<th>Joy</th>
<th>Sadness</th>
<th>Surprise</th>
<th>Trust</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>0.30</td>
<td>0.03</td>
<td>0.00</td>
<td>0.00</td>
<td>0.06</td>
<td>0.03</td>
<td>0.42</td>
<td>0.09</td>
</tr>
<tr>
<td>Anticipation</td>
<td>0.03</td>
<td>0.00</td>
<td>0.03</td>
<td>0.31</td>
<td>0.13</td>
<td>0.06</td>
<td>0.28</td>
<td>0.09</td>
</tr>
<tr>
<td>Disgust</td>
<td>0.13</td>
<td>0.00</td>
<td>0.28</td>
<td>0.15</td>
<td>0.21</td>
<td>0.00</td>
<td>0.23</td>
<td>0.00</td>
</tr>
<tr>
<td>Fear</td>
<td>0.05</td>
<td>0.00</td>
<td>0.00</td>
<td>0.51</td>
<td>0.16</td>
<td>0.02</td>
<td>0.15</td>
<td>0.02</td>
</tr>
<tr>
<td>Joy</td>
<td>0.00</td>
<td>0.03</td>
<td>0.08</td>
<td>0.07</td>
<td>0.51</td>
<td>0.02</td>
<td>0.27</td>
<td>0.02</td>
</tr>
<tr>
<td>Sadness</td>
<td>0.12</td>
<td>0.03</td>
<td>0.05</td>
<td>0.15</td>
<td>0.12</td>
<td>0.26</td>
<td>0.24</td>
<td>0.03</td>
</tr>
<tr>
<td>Surprise</td>
<td>0.01</td>
<td>0.00</td>
<td>0.04</td>
<td>0.05</td>
<td>0.14</td>
<td>0.04</td>
<td>0.66</td>
<td>0.05</td>
</tr>
<tr>
<td>Trust</td>
<td>0.09</td>
<td>0.00</td>
<td>0.06</td>
<td>0.03</td>
<td>0.25</td>
<td>0.03</td>
<td>0.38</td>
<td>0.16</td>
</tr>
</tbody>
</table>

(a) Joint representation

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Anger</th>
<th>Anticipation</th>
<th>Disgust</th>
<th>Fear</th>
<th>Joy</th>
<th>Sadness</th>
<th>Surprise</th>
<th>Trust</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>0.48</td>
<td>0.03</td>
<td>0.03</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.42</td>
<td>0.03</td>
</tr>
<tr>
<td>Anticipation</td>
<td>0.06</td>
<td>0.00</td>
<td>0.15</td>
<td>0.25</td>
<td>0.09</td>
<td>0.03</td>
<td>0.31</td>
<td>0.09</td>
</tr>
<tr>
<td>Disgust</td>
<td>0.03</td>
<td>0.00</td>
<td>0.54</td>
<td>0.17</td>
<td>0.05</td>
<td>0.00</td>
<td>0.18</td>
<td>0.05</td>
</tr>
<tr>
<td>Fear</td>
<td>0.03</td>
<td>0.00</td>
<td>0.14</td>
<td>0.53</td>
<td>0.05</td>
<td>0.00</td>
<td>0.18</td>
<td>0.05</td>
</tr>
<tr>
<td>Joy</td>
<td>0.02</td>
<td>0.02</td>
<td>0.10</td>
<td>0.06</td>
<td>0.55</td>
<td>0.00</td>
<td>0.19</td>
<td>0.06</td>
</tr>
<tr>
<td>Sadness</td>
<td>0.06</td>
<td>0.00</td>
<td>0.09</td>
<td>0.24</td>
<td>0.03</td>
<td>0.32</td>
<td>0.22</td>
<td>0.03</td>
</tr>
<tr>
<td>Surprise</td>
<td>0.03</td>
<td>0.00</td>
<td>0.04</td>
<td>0.09</td>
<td>0.04</td>
<td>0.04</td>
<td>0.19</td>
<td>0.00</td>
</tr>
<tr>
<td>Trust</td>
<td>0.03</td>
<td>0.00</td>
<td>0.03</td>
<td>0.00</td>
<td>0.31</td>
<td>0.03</td>
<td>0.16</td>
<td>0.44</td>
</tr>
</tbody>
</table>

(b) Fusion

Figure 5.3: Confusion matrix based on (a) joint representation and (b) fusion results on the VideoEmotion dataset.
Figure 5.4: Prediction accuracies. SentiBank is the attribute feature proposed in [1]. E-MDBM-V and E-MDBM-A represent the joint representation generated by our proposed E-MDBM using only the visual or auditory signals as inputs. Similarly, E-MDBM-VA indicates the joint representation using both visual and auditory modalities.

In addition, only visual features are used in SentiBank, AF and E-MDBM-V. Our model outperforms SentiBank and AF by 7.7% and 5.6% in accuracy respectively. This is because that our model embeds the information from all the three modalities during unsupervised model training. Thus the correlations between visual modality and other two modalities can be jointly represented, especially the textual modality which helps to explore the semantics in the videos. Comparing to SentiBank, where the extracted semantics are limited to 1,200 predefined ANPs, our model trained on wild Web data is expected to capture more complex semantics. AF, which comprises hand-tuned features dedicated for art photos, still performs reasonably well on the web video domain. This basically gives clue to the correlation between art-based features and emotions. Interestingly, AF even performs slightly better than SentiBank that predefines the set of ANPs for emotion description. E-MDBM-V, which learns features directly from examples while considering multi-modality correlation, has better capacity in dealing with diversities in user-generated videos compared to SentiBank and AF.

We can also observe that the auditory information seems less effective comparing to visual information. However, it works better in “Disgust” and “Sadness” categories, where the visual information cannot provide enough clues for emotion detection. For instance, there is a video showing a cat, which is annotated as “Dis-
Table 5.3: Prediction accuracies of the state-of-the-arts on VideoEmotion [5]. The notations V., Au., and At. represent visual, auditory and attribute features respectively.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>0.345</td>
<td>0.388</td>
<td>0.294</td>
<td>0.303</td>
<td><strong>0.549</strong></td>
<td>0.527</td>
<td>0.515</td>
<td>0.527</td>
</tr>
<tr>
<td>Anticipation</td>
<td><strong>0.082</strong></td>
<td>0.031</td>
<td>0.067</td>
<td>0</td>
<td>0.028</td>
<td>0.067</td>
<td>0.030</td>
<td>0.067</td>
</tr>
<tr>
<td>Disgust</td>
<td>0.237</td>
<td>0.298</td>
<td>0.267</td>
<td>0.282</td>
<td>0.399</td>
<td>0.381</td>
<td>0.308</td>
<td>0.438</td>
</tr>
<tr>
<td>Fear</td>
<td>0.414</td>
<td>0.435</td>
<td>0.395</td>
<td>0.509</td>
<td>0.396</td>
<td>0.484</td>
<td><strong>0.648</strong></td>
<td>0.471</td>
</tr>
<tr>
<td>Joy</td>
<td>0.398</td>
<td>0.396</td>
<td>0.442</td>
<td>0.508</td>
<td>0.480</td>
<td>0.557</td>
<td>0.567</td>
<td>0.484</td>
</tr>
<tr>
<td>Sadness</td>
<td>0.106</td>
<td>0.063</td>
<td>0.217</td>
<td>0.265</td>
<td>0.289</td>
<td>0.274</td>
<td>0.229</td>
<td>0.208</td>
</tr>
<tr>
<td>Surprise</td>
<td>0.735</td>
<td>0.688</td>
<td>0.666</td>
<td>0.675</td>
<td>0.746</td>
<td>0.802</td>
<td><strong>0.861</strong></td>
<td>0.767</td>
</tr>
<tr>
<td>Trust</td>
<td>0.052</td>
<td>0.061</td>
<td>0.136</td>
<td>0.156</td>
<td>0.311</td>
<td>0.327</td>
<td>0.250</td>
<td>0.287</td>
</tr>
<tr>
<td>Overall</td>
<td>0.352</td>
<td>0.359</td>
<td>0.371</td>
<td>0.494</td>
<td>0.451</td>
<td>0.484</td>
<td>0.501</td>
<td>0.463</td>
</tr>
</tbody>
</table>

The visual appearance actually conveys no emotional information. On the other hand, the background music, which is very sharp and noisy, actually makes people feel uncomfortable and disgust. The same situation exists in the “Sadness” videos. There are only several common objects shown in the video, such as faces and people hugs, whereas, woeful music is used as background. In short, missing of modality will degrade the performance even using our proposed model. However, the joint representation can somehow capture the correlations between different modalities, and is a good compensation when certain modality is not available.

Comparison with state-of-the-arts. We compare our model with the simplified version in [2]. For fair comparison, we extend the model in [2] to handle three modalities by adding a new pathway for textual input. This model is named as MDBM. Same training data and settings described in Section 5.4.1 are employed for learning MDBM. We also show the results reported in [5] here. These results can be considered as the state-of-the-arts as they are produced through fine tuning on the dataset. In [5], various auditory (Au.) features, visual (V.) features, attribute (At.) features, and their combinations are evaluated. Table 5.3 shows the best one in each kind of the feature. Furthermore, the combinations of our joint representation and other features are also included. In addition, the performance based on the art features (AF) [4] is also reported.

We can see that E-MDBM consistently outperforms MDBM either evaluated individually or combined with other features. Specifically, E-MDBM leads to 9% performance improvement over MDBM. This indicates that our design can better preserve the unique property of each visual feature during the learning in visual pathway by splitting the architecture into several sub-pathways, each of which
corresponds to one feature. In contrast, all the features are concatenated into one feature vector in MDBM, where some visual features may be overwhelmed during the learning. However, V.+Au. performs better than E-MDBM. This is probably because the pre-training of our model is performed on Flickr images, while V.+Au. is tuned on the Web videos in VideoEmotion dataset. The domain gap may influence the learned joint representation. As stated in [5], this may also cause the performance degradation of attribute features (e.g., SentiBank) which are extracted using concept classifiers learned on Web images. Despite the domain gap, E-MDBM consistently achieves better performance than SentiBank, either when being utilized individually or fused with other features. While not performing better than hand-crafted features, E-MDBM can complement these features well. When average lately fused with features in [5], an improvement of 11.09% is attained. Further fusion with attribute features (i.e., SentiBank, Classemes and ObjectBank), an accuracy of 0.511 is attained. The degree of improvement introduced by E-MDBM is greater than that can be offered by MDBM. Figure 5.3(b) shows the confusion matrix based on fusion results of E-MDBM. Compared with Figure 5.3(a), four categories, especially “Trust”, become less confused by “Surprise” after fusion. Nevertheless, the performance for “Anticipation” remains poor.

5.4.3 Sentiment Analysis on Twitter Messages

To avoid the impact of domain shift, we further conduct experiments on “ImageTweets” dataset [1], which includes 596 text-image Twitter messages. In this dataset, only textual and visual modality are available. These messages are manually assigned to either positive or negative sentiment based on affection expressed in the text-image pairs.

We compare our model with several state-of-the-art methods used in [1]. Besides the early fused low-level visual features (“Visual”) and attribute feature (“SentiBank”) extracted from images, a lexicon-based approach (“Lexicon”) used on textual analysis is also selected as baseline. The text information is represented using the sentiment scores of words obtained from SentiStrength [3]. The art features (AF) [4] is also adopted as one baseline. The classifiers for textual feature and visual features are Naive Bayes classifier and logistic regression model respectively. MDBM [2] is utilized on this dataset with visual+textual input. For our proposed E-MDBM, we consider three features E-MDBM-V, E-MDBM-T and E-
Figure 5.5: Prediction accuracies on the ImageTweets. E-MDBM-V and E-MDBM-T are the classifiers trained on the joint representation generated by using only the visual or textual information through E-MDBM. E-MDBM-VT is trained on the joint representations over both visual and textual modalities. MDBM represents the joint representations over both visual and textual modalities but based on the architecture proposed in [2]. Meanwhile, SentiBank represents the classifiers trained on the scores of the concepts classifiers in [1]. Lexicon represents the Naive Bayes classifiers trained based on SentiStrength [3]. AF represents the classifiers trained on the art features proposed in [4].
MDBM-VT corresponding to visual input, textual input and visual+textual input respectively. For fair comparison, logistic regression model is used upon these joint representations.

In [1], the dataset is equally split into five subsets. In our experiment, each classifier is trained on four subsets and tested on the other. This process is repeated five times. Figure 5.5 shows the average results. We first consider the case of single modality input. We can see that Lexicon performs much worse than other methods. This is not surprising since Twitter messages are usually sparse and lack of emotion signals. In comparison, with the same input, our joint representation E-MDBM-T achieves 43.49% improvement over Lexicon. Again, this demonstrates that the common space embedded in the E-MDBM preserves the correlations from multiple modalities with different emotional signals. Thus the sparsity problem can be addressed using the information from other modalities by mapping from textual input to the joint space. For the visual modality, different from the results in Table 5.3, SentiBank and E-MDBM-V perform much better than low level visual features (Visual) on image dataset. This is because both our model and attribute feature represent the semantics in the images, which can narrow down the gap between low-level features and high level human perceptions. In addition, AF achieves a similar performance to SentiBank, indicating that the Tweet images also partially follow the artistic principles. Again, E-MDBM-V improves over SentiBank and AF by 4.14% and 3.11% in accuracy respectively. We then compare the performances of different methods using multiple modalities. For comparison, we also show the result of lately fused Lexicon and SentiBank (SentiBank+Lexicon) features. We can see that E-MDBM-VT exhibits the best results, which leads to 8% and 5% improvements over SentiBank+Lexicon and MDBM respectively. Finally, similar to the conclusion made on VideoEmotion dataset, the performances of E-MDBM-V and E-MDBM-T are worse than that of E-MDBM-VT, which also indicates that there are some noises in the generated missing modalities.

5.5 Experiment: Retrieval

This section experiments the feasibility of the proposed model for video retrieval (Section 5.5.1) and cross-media retrieval (Section 5.5.2). A new dataset including 1,139 reference videos was constructed using 23 ANPs in [1] as queries. Both
Table 5.4: Mean average precision@20 of text-based, video-based and multimodal query for retrieving emotional videos.

<table>
<thead>
<tr>
<th>Category</th>
<th>Text-based Query</th>
<th>Video-based Query</th>
<th>Multimodal Query</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>0.222</td>
<td>0.345</td>
<td>0.340</td>
</tr>
<tr>
<td>Anticipation</td>
<td>0.202</td>
<td>0.437</td>
<td>0.384</td>
</tr>
<tr>
<td>Disgust</td>
<td>0.389</td>
<td>0.446</td>
<td>0.492</td>
</tr>
<tr>
<td>Fear</td>
<td>0.417</td>
<td>0.423</td>
<td>0.436</td>
</tr>
<tr>
<td>Joy</td>
<td>0.401</td>
<td>0.490</td>
<td>0.528</td>
</tr>
<tr>
<td>Sadness</td>
<td>0.337</td>
<td>0.492</td>
<td>0.422</td>
</tr>
<tr>
<td>Surprise</td>
<td>0.508</td>
<td>0.554</td>
<td>0.640</td>
</tr>
<tr>
<td>Trust</td>
<td>0.333</td>
<td>0.400</td>
<td>0.443</td>
</tr>
<tr>
<td>Overall</td>
<td>0.353</td>
<td>0.450</td>
<td>0.470</td>
</tr>
</tbody>
</table>

videos and their surrounding metadata are downloaded from YouTube. With this dataset, we can perform Web video retrieval using different types of queries. In specific, we randomly select 50 videos covering all the eight emotions in [5] as queries. In some of the queries, the content and emotion in different modalities are not always aligned. Figure 5.6 shows some examples of queries. The selected 23 ANPs include 18 emotion words\(^1\) and 16 general concept words\(^2\). We first annotate all the test and query videos for the 34 words. A video is considered to be relevant to the query if they share at least one emotion category or concept. In other words, two videos can be semantically or emotionally relevant. In this way, we can generate ground-truth for each query. Note that as emotion words are extracted from ANPs, their categories are inferred from which the ANPs belong to.

5.5.1 Video Retrieval

Three sets of experiments are conducted by using the text, video and multimodal (text+video) queries. For each query, a joint representation is extracted using the proposed E-MDBM. Note that there are missing modalities for text or video queries. For the reference videos, we assume that all the three modalities are

---

\(^1\)List of emotion words: cold, broken, fantastic, curious, classic, fat, crazy, lazy, creepy, haunted, clear, bright, natural, heavy, dirty, adorable, shiny, tasty

\(^2\)List of concept: morning, chair, fence, architecture, bird, spider, cat, tree, castle, moon, spring, rain, dog, star, food, body
Figure 5.6: Examples of video queries. The caption for the first example (top-left) describes disable kitten, but the video expresses a “joy” emotion with audio-visual effect. The two videos in the second row show counter examples. The video on the left has a spider but is not mentioned by caption, as opposed to the video on the right where spider is mentioned in caption but dose not appear in video. The query (top-right) expresses consistent emotion and semantics across visual, auditory and textual modalities.
available when extracting the joint representations. The baselines used in the experiments depend on the modality of queries. For text query, we compute Jaccard coefficient between the word count vector of a query and the surrounding text of a reference video. For video query, we select two baseline methods that utilize visual and auditory information by fusing low level visual feature and SentiBank respectively with auditory feature. For multimodal query, the two baselines are further augmented using textual feature. In this way, for a given type of query, the compared methods leverage same set of input features.

Table 5.4 shows the results of video retrieval on eight different categories of emotions in terms of mean average precision (MAP). E-MDBM achieves consistently better performances than the baselines in all three types of queries. As baseline methods consider only matching the modalities of same type during similarity measure, reference videos with the searched content or emotion exists in a modality different from the type of query cannot be retrieved. For example, although the top-left video in Figure 5.6 shows a “joy” emotion with a dog befriends a cat, text-only query is misled by the word “disable”. For this particular example, the performance is poor even for multimodal query with late fusion strategy. Similarly for the two queries shown in the second row of Figure 5.6, where there are mismatches between the concepts in captions and video content. Late fusion of multiple modalities helps little for these example queries. In contrast, by non-linearly projecting all the reference videos into a joint space, E-MDBM has generated features that inherently capture the complex relationship among different modalities of videos. Hence, cross-modal matching between queries and reference videos are implicitly performed during retrieval. For the examples in Figure 5.6, E-MDBM shows improvement in large margin, for example, the AP for the top-left query achieved by V.+Au.+Te. and SentiBank+Au.+Te. are 0.193 and 0.247 respectively. In contrast, our joint representation exhibits much better AP with 0.394. Finally, it is worth mentioning that using E-MDBM, each video is represented as a feature in 4,096 dimensions, in contrast to 28,965 dimensions of hand-crafted features. The compact feature representation will greatly save time and space in video retrieval.

5.5.2 Cross-Modal Retrieval

We further evaluate our method on cross-modal retrieval. Different from the experiment in 5.5.1, where the feature extracted for database videos adopts all the
Table 5.5: Cross-modal retrieval: Mean average precision @ 20 on four different types of queries against four different versions of datasets.

<table>
<thead>
<tr>
<th>Modality</th>
<th>DB-T</th>
<th>DB-V</th>
<th>DB-A</th>
<th>DB-VA</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-MDBM-T</td>
<td>-</td>
<td>0.366</td>
<td>0.371</td>
<td>0.368</td>
</tr>
<tr>
<td>E-MDBM-V</td>
<td>0.437</td>
<td>-</td>
<td>0.358</td>
<td>-</td>
</tr>
<tr>
<td>E-MDBM-A</td>
<td>0.351</td>
<td>0.365</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>E-MDBM-VA</td>
<td>0.430</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 5.6: Cross-modal retrieval: Mean average precision@20 for eight emotion categories under different scenarios. The best performance for each category is highlighted.

<table>
<thead>
<tr>
<th>Category</th>
<th>T→V+A</th>
<th>T→V</th>
<th>T→A</th>
<th>A→T</th>
<th>A→V</th>
<th>V→A</th>
<th>V→T</th>
<th>V+A→T</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>0.222</td>
<td>0.224</td>
<td>0.280</td>
<td>0.233</td>
<td>0.223</td>
<td>0.253</td>
<td>0.376</td>
<td>0.376</td>
</tr>
<tr>
<td>Anticipation</td>
<td>0.205</td>
<td>0.200</td>
<td>0.218</td>
<td>0.200</td>
<td>0.200</td>
<td>0.202</td>
<td>0.246</td>
<td>0.200</td>
</tr>
<tr>
<td>Disgust</td>
<td>0.383</td>
<td>0.384</td>
<td>0.394</td>
<td>0.390</td>
<td>0.383</td>
<td>0.385</td>
<td>0.475</td>
<td>0.473</td>
</tr>
<tr>
<td>Fear</td>
<td>0.374</td>
<td>0.356</td>
<td>0.473</td>
<td>0.401</td>
<td>0.356</td>
<td>0.345</td>
<td>0.667</td>
<td>0.656</td>
</tr>
<tr>
<td>Joy</td>
<td>0.475</td>
<td>0.472</td>
<td>0.411</td>
<td>0.386</td>
<td>0.471</td>
<td>0.394</td>
<td>0.456</td>
<td>0.450</td>
</tr>
<tr>
<td>Sadness</td>
<td>0.333</td>
<td>0.333</td>
<td>0.333</td>
<td>0.333</td>
<td>0.333</td>
<td>0.337</td>
<td>0.380</td>
<td>0.372</td>
</tr>
<tr>
<td>Surprise</td>
<td>0.500</td>
<td>0.500</td>
<td>0.504</td>
<td>0.519</td>
<td>0.500</td>
<td>0.504</td>
<td>0.560</td>
<td>0.566</td>
</tr>
<tr>
<td>Trust</td>
<td>0.333</td>
<td>0.333</td>
<td>0.333</td>
<td>0.333</td>
<td>0.333</td>
<td>0.444</td>
<td>0.333</td>
<td>0.333</td>
</tr>
</tbody>
</table>
three modalities, cross-modal retrieval assume that query and the candidate data have different form of media types. Many applications can benefit from the cross-modal retrieval. For example, given a video without text description, the retrieved textual documents are helpful for automatically understanding the videos. Comparing to traditional semantic concept detection, which aims at recognizing some general concepts from visual appearance or auditory signals, the generated description using cross-modal retrieval would be more comprehensive.

Using the same dataset as Section 5.5.1, we generate four different versions of datasets, where each with one or two modalities of reference videos being purposely omitted. The four datasets are named as DB-X, with X-only modality being generated by the joint representation, where X includes T (text), V (visual), A (audio) and VA (visual-audio) modalities. During experiments, E-MDBM-X is experimented to search on a dataset without modality X. For example E-MDBM-T (text-only query) is searched against the dataset DB-V where the text and audio signals of all videos are ignored. Table 5.5 summarizes the results of cross-modal retrieval for 50 queries. As can be observed, the performances are very encouraging. For example, E-MDBM-T can achieve MAPs of 0.366 and 0.371 on DB-V and DB-A respectively when metadata of reference videos are ignored. These results are better than text-based query using word count (MAP=0.353) when the metadata are involved in similarity comparison, as shown in Table 5.4. Using E-MDBM-V (visual-only query) against DB-T can attain MAP of 0.437, which is fairly impressive given that only the metadata of reference videos are kept for similarity measure. The result is close to SentiBank (AP=0.446), which compares similarity with reference videos directly based on visual modality.

We further detail the results for each emotion category in Table VI, where every column corresponds to a different cross-modal retrieval scenario. For example, T→V+A means the use of text-based query for searching against videos with only visual and audio modalities. Interestingly, query-by-visual often exhibits better performance even though the visual modality of reference videos is not considered. In addition, visual seems to be correlated better with text than audio modality, resulting in better performance in 7 out of eight categories for V→T. The only exception is the category “Trust”. This is because many videos expressing the trust emotion in this dataset are about trust test with laughing and cheering sounds. Comparing Table 5.6 with Table 5.4, which can be considered as the up-bound of cross model retrieval, there is still a performance gap. However, considering that we are matching the data from different modalities with different statistic
properties, basically the results in Table 5.5 can demonstrate the ability of our method for modeling the joint representations.

5.6 Summary

We have presented a deep model for learning multimodal signals coupled with emotions and semantics. Particularly, we propose a multi-pathway DBM architecture dealing with low-level features of various types and more than twenty-thousand dimensions, which is not previously attempted to the best of our knowledge. The major advantage of this model is on capturing the non-linear and complex correlations among different modalities in a joint space. The model enjoys peculiarities such as learning is unsupervised and can cope with samples of missing modalities.

Compared with hand-crafted features, our model generates much more compact features and allows natural cross-modal matching beyond late or early fusion. As demonstrated on ImageTweets datasets, the features generated by mapping single-modality samples (text or visual) into the joint space consistently outperform hand-crafted features in sentiment classification. In addition, we show the complementary between deep and hand-crafted features for emotion prediction on VideoEmotion dataset. Among the eight categories of emotion, nevertheless, the categories “anticipation” and “surprise” remain difficult either with learnt or hand-tuned features. For video retrieval, our model shows favorable performances, convincingly outperforms hand-crafted features over different types of queries. Encouraging results are also obtained when applying the deep features for cross-modal retrieval, which is not possible for hand-crafted features. Compared to SentiBank, our model has the edge of not limiting to a predefined set of vocabularies. Hence, the learning is fully generative and the model is more expressive, as we show in the experiments that our model is able to perform better than SentiBank in both classification and retrieval tasks. Finally, our model also consistently outperforms the early version of MDBM [2] in all the experiments conducted in this chapter.
CHAPTER 6

CONSTELLATION BROWSER FOR EXPLORATORY SEARCH

Video searching experience can be imagined as “searching the shining stars in the galaxy”. Looking up into the sky, one may be confused and overwhelmed by the clouds of stars. The shining stars are more easily located, nevertheless, if the galaxy could be organized into different constellations based on certain knowledge. Inspired by this fact, we believe that user browsing experience can be improved by providing functionalities that expose different patches of perceived signals for efficient navigation, similar to plotting the constellation in the galaxy.

This chapter integrates face, sentiment and emotion for exploratory search of opinion-oriented videos. Different from commercial search engines which return a ranked list of videos, we present the search results as a graph with three different kinds of hyperlinks connecting different segments of videos. The hyperlinks are established based on results of face labeling, emotion categorization and topic relatedness, aiming to facilitate exploratory search for users performing “casual” search without a specific search goal. As the survey reported in [118], 37% of users perform browsing casually without any specific videos in mind. List-wise presentation of videos tells very little about the relationship among videos and hence is less helpful for casual type of browsing behavior. By hyperlinking videos based on the perceived signals underlying content, users can more easily navigate the result and switch the search focus based on the information encountered.

We consider three kinds of hyperlinks: topic, emotion and celebrity links, respectively based on the results on text, affect and face analysis. Given an opinion question described in text, sentiment-oriented clips are retrieved. These videos are further expanded by topic links to connect to factoid clips, which provide relevant knowledge in understanding the background of different opinions. Emotion links take place by connecting videos with the same emotion category, providing a gist of general opinions about the question of interest. In some cases, factoid clips expressing strong emotion are usually videos describing milestone events. For example, an critical moments for “presidential election” is the winning moment, which usually
consists of “happy guy” for celebration. Finally, celebrity links connect clips based on person identity and face appearance, providing a different navigation path from the perspective of person as subject of interest. News topic are usually evolved based on what the celebrities have said or done and the “celebrity link” enables users to overview the topic following the trace of the celebrity that they are interested in. For example, along the links based on the “Cola boy” in “Sichuan earthquake”, users can easily locate the important events on him, such as rescue in earthquake, the amputation, and college life in Shanghai. Based on these three kinds of links, users can easily zoom in to a specific event by topic and emotion links or zoom out for a human subject by celebrity link.

This chapter presents a developed system named “Constellation”, by mining and linking different perceived signals for construction of constellation for novel exploratory search. Guided with different means of hyperlinks, users can quickly grasp the gist of different events by browsing the videos intelligently rather than go through the video list in successive order.

6.1 Technology

This section outlines the technology for video searching and constellating in response to users’ opinion questions. As shown in Figure 6.1, the system building is split into two stages, i.e., offline and online stages. We start by introducing the procedures in the offline stage, including “Deep Match” learning for opinion QA (Section 4.3), “celebrity inverted index” (Section 3.1) for constructing celebrity links and topic and emotion link lookup table (Section 5.1) for constructing topic and emotion links respectively. In the online stage, the video clips are retrieved based on the “Deep Match” model and then organized with the three kinds of links. After browsing through the retrieved videos, users can refine their queries and search again.

6.1.1 Offline Preprocessing

All the procedures in the preprocessing stage are carried out in parallel for large scale video processing.
Figure 6.1: Framework of Constellation Browser. The online searching is conducted based on the “Deep Match” model. The celebrity, topic and emotion links in constellation presentation are constructed based on the celebrity inverted index, topic and emotion link lookup tables respectively. After exploring the searching result, users can refine queries and search again.
6.1.1.1 Video Processing for Link Construction

The videos in the candidate dataset are first split into event-based clips using speaker diarization [81] and TextTiling [119]. As discussed in Section 4.1.1, speaker diarization makes sure the integrity of comments. Meanwhile, TextTiling is adopted to ensure the integrity of context, which may combine several consequent speaker diarization segments together based on the word distribution in transcripts. For example, the discussion between two persons towards one event should be treated as one segment. However, we observed that there is usually one single person talking all the time in the documentary videos. Hence, the speaker diarization segments are further split when the interval is larger than 10 seconds in the transcripts. The celebrity, topic and emotion links among these video clips as shown in Figure 6.1 are constructed as follows.

- **Celebrity inverted index.** The celebrities in the candidate video clips are detected based on the algorithms proposed in Section 3.2.3. The celebrities appeared in an video clip will be stored to facilitate users navigating. For example, displaying the celebrities when user hovers over a video node may help users to determine watching the video or not. To speed up the link construction, the celebrities in the candidate clips are further inverted indexed.

- **Topic link lookup table.** Word2Vec [19, 20] has achieved great success in measuring syntactic and semantic word similarities. In addition, the word vector representation also works well for phrases and short text with simple algebraic operations, for example representing a phrase by adding all the word vectors in it [19]. In the proposed system, each video clip is represented by adding all the word vectors in its transcripts. For word vector generation, we finetune the Google News model\(^1\) with the candidate clips. The topic links among the video clips are constructed based on the cosine similarities of their vector representations. By setting a threshold, we convert the topic links into a binary lookup table, which facilitates the link construction.

- **Emotion link lookup table.** In Section 5.5, the multimodal DBM model (Section 5.4.1) has demonstrated its effectiveness in measuring the emotion similarities among videos. In the proposed system, the emotion links are

\(^1\)https://code.google.com/p/word2vec/
constructed based on the cosine similarities of the multimodal DBM representation. Similar to topic links, we also index emotion links with a lookup table for speeding up browsing.

### 6.1.1.2 Deep Match Learning for Opinion QA

For “Deep Match” learning, we extend from topic specific to topic unconstrained QA modeling. In Section 4.3, the training QA pairs are crawled based on the topics. In this chapter, we crawl more QA pairs based on Yahoo!Answers categories rather than topics. Since the QA pairs in Yahoo!Answers almost covers all kinds of information in our life, ranging from arts to technology, we believe that the crawled QA pairs can provides a comprehensive understanding about the hot spots currently. In addition, the proposed “Deep Match” learning can be incrementally learnt. Hence, the model can keep updating with the hot events. We adopt similar deep structure as described in Section 4.3.1 but expanding the number of neurons in p-layerI, p-layerII and committee layer to cover the diversity of topics. In the proposed system, 1,000 and 300 events are learnt and each event is represented by three neurons. Therefore, there are 3,000 and 900 neurons in p-layerI and p-layerII respectively. The committee layer is expanded to 200 neurons.

Based on the methods proposed in sections 4.1 and 4.2, we classify the candidate video clips into sentiment and factoid clips. In the sentiment clips, a person will present his opinions towards a specific event. Meanwhile, the factoid clips mainly introduce the background knowledge about a specific event. Note that, the factoid clip may also express emotions. For example, the factoid video clip v4 in Figure 6.2, which introduces the “tofu-dregs schoolhouses”, expresses the emotion of anger towards the corruption in China.

### 6.1.2 Online Constellation Searching

Online search starts with user giving an opinion question. Constellation will be constructed on the fly as followings. First, sentiment clips are retrieved based on deep match model. Second, using the lookup table of topic links, relevant factoid clips are further returned and linked to the sentiment clips. Finally, emotion and celebrity links are established among them based on lookup table and inverted celebrity index. Figure 6.2 shows an example of constellation. The relationships
Figure 6.2: Example Constellation on the “Cola Boy” in Sichuan Earthquake. The surrounding rectangles of nodes represent the node types: green for positive sentimental clips, red for negative sentimental clips and black for factoid clips. The dotted and solid black edges represent celebrity and topic links respectively. Meanwhile, the colored edges represent eight different emotions: red for anger, bright green for joy, dark green for fear, light blue for surprise, dark blue for sadness, pink for disgust, dark orange for anticipation, light yellow for trust. In this example, only two kinds of emotions, i.e., anger and joy, are shown.
between clips are visualized with different patterns and colors. The solid and dotted lines indicate the topic and celebrity links respectively. The emotion links are colored according to the Plutchik’s wheel. Each sentiment clip is also bounded with either green or red box, representing positive and negative sentiment respectively. The intensity of color shows the degree of sentiment. The bounding box of factoid clips are in black color so as to distinguish from sentiment clips.

Figure 6.2 shows a constellation for the question “What do you think of Cola boy (Xue Xiao) in Sichuan earthquake?” The clips $v_1$–$v_3$ deliver opinions of “tofu-dregs schoolhouses” from different parties, including the parents who lost their children. These videos form a clique for expressing the same mood of emotion. The factoid clip $v_1$, which is a documentary introducing the event “collapse of schoolhouses”, is connected to $v_1$–$v_3$ via a topic link. The linkages among these four videos demonstrate an example of how user can navigate by traversing between an event and the opinions about the event. The clip $v_4$ is further linked to $v_5$, which is a factoid clip reporting “Cola boy” who is buried in the ruins of schoolhouses. Further down $v_5$ is $v_7$, which reported the moment when “Cola boy” and his classmates are rescued. The link is colored in green, signifying the moment of joy emotion. The clip $v_6$, which reported public response and support for the rescue, is also connected to $v_5$ with green. The clips $v_8$ to $v_{11}$ are about the personal life of “Cola boy” before and after the earthquake. They are (in)directly connected to $v_5$ via topic and celebrity links with dotted lines, providing clue that users can further dig down to know more about “Cola boy”. Assuming a user starts navigation from $v_5$, the diversity of links from $v_5$ ideally will explore the user to three different aspects of the story.

6.2 User Study

Figure 6.3 shows the web version of Constellation. The interface is Google like with a search bar, and is simple and easy to use. Constellation is displayed in a window that simulates the search of shining stars using a telescope. When mouse hovers over the clips, captions such as name entities, celebrities and emotions will be popped out. Using this interface, we conduct user study to assess the utility of Constellation.
6.2.1 Dataset

The experiment is conducted on LNK dataset provided by TRECVID 2015 for video hyperlinking task\(^2\). The dataset contains around 3,000 hours of BBC videos, including 600 different TV programs ranging from architecture history to natural disasters. By our system, the videos are segmented into 100,917 clips. Among them, 21,335 clips (21.14\%) are classified as sentiment oriented. Stanford name entity recognizer\(^3\) was applied for name entities extraction from speech transcripts. By matching the extracted entities with the entries in DBpedia and Freebase, a total of 47 celebrities are identified and name-face association is performed for these celebrities. The thresholds for establishing topic and emotion links are respectively 0.85 and 0.90 by cosine similarity.

To train the deep match model for QA matching, more than two millions of QA pairs covering 767 categories are crawled from Yahoo! Answers. The system is run on a server with 24 CPUs and 60G of main memory. By the current setup, online search costs around 1 second and building of constellation further takes another 0.5 seconds. In other words, a user needs to wait for around 1.5 second from issuing a question to before interacting with the constellation.


\(^3\)http://nlp.stanford.edu/software/CRF-NER.shtml
Table 6.1: The four opinion questions posted of user studies.

<table>
<thead>
<tr>
<th>Question</th>
<th>#video clips</th>
<th>#celebrity links</th>
<th>#topic links</th>
<th>#emotion links</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. What do you think about the collapse of schoolhouses in Sichuan earthquake?</td>
<td>67</td>
<td>9</td>
<td>62</td>
<td>15</td>
</tr>
<tr>
<td>2. What do you think about the rescue in Nargis cyclone?</td>
<td>55</td>
<td>4</td>
<td>51</td>
<td>21</td>
</tr>
<tr>
<td>3. What do you think about the traditional architecture in British?</td>
<td>73</td>
<td>5</td>
<td>68</td>
<td>8</td>
</tr>
<tr>
<td>4. What do you think about the protection of the wild animals?</td>
<td>81</td>
<td>3</td>
<td>76</td>
<td>31</td>
</tr>
</tbody>
</table>

6.2.2 Evaluation and Result

The user study aims to compare constellation and list-wise browsing. Ten evaluators, five males and females, are recruited for the evaluation. The evaluators come from different education backgrounds: computer science (7), chemistry (1), business (1) and art (1). Their ages range from 23 to 32, and average at 27. During the evaluation, each subject is given 4 opinion questions of which two of them should be presented with constellation and the other two with ranked list. Table 6.1 lists the four questions, and the numbers of videos and links for each question. To minimize the carryover effect, questions and presentations were assigned randomly. Specifically, each subject was requested to use constellation to answer the first two questions, followed by ranked list for the remaining two questions, or vice versa. Three criteria are defined for evaluation:

- **Precision**: Are the retrieved sentiment clips relevant to the opinion question?
- **Engagement**: How useful the browsing system is in providing guidance for understanding opinions in the sentiment clips?
- **Acceptance**: Will the browsing system lead to better user experience?

Since the relatedness of the sentimental clips to the questions could be subjective, Precision is evaluated by each subject on every question. Meanwhile, Engagement and Accuracy are for evaluating the overall user experience of these two presentations, which are evaluated after all the four questions. Here the precision
Table 6.2: The average scores and standard deviation on Precision, Engagement and Acceptance for Constellation and list-wise presentation.

<table>
<thead>
<tr>
<th></th>
<th>Constellation</th>
<th>Ranked List</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>0.61 ± 0.22</td>
<td></td>
</tr>
<tr>
<td>Engagement</td>
<td>4.00 ± 0.67</td>
<td>2.00 ± 0.82</td>
</tr>
<tr>
<td>Acceptance</td>
<td>3.40 ± 0.97</td>
<td>1.90 ± 0.74</td>
</tr>
</tbody>
</table>

is calculated based on the top-5 retrieved sentiment clips. The Engagement and Acceptance are rated in the scale of 0 to 5, with 5 being the best and 0 being the worst. Table 6.2 shows the evaluation score and standard deviation averaged over all the evaluators. The detailed precision for each question is listed in Figure 6.4. Based on the result in terms of precision, we can see that “Deep Match” model achieves acceptable performance even in topic-unconstrained situation. Question 2 and 4 suffer from lower precision due to lack of appropriate training data. For example, the retrieved clips for Q2 also include opinions about the rescue in other disasters, such as earthquake. As for the overall user experience, as observed in Table 6.2, evaluators prefer the constellation much more than list-wise presentation. The Engagement reflects that the retrieved opinions can be better understood in constellation with the provided different kinds of links. However, we also note that the Acceptance score is around the average level, although higher than that of ranked list view.

Based on the verbal feedback, the average performance in Acceptance is mainly attributed to the layout of the constellation presentation. When there are excessive number of clips and links, the force directed layout is too dense, making the selection of clips and navigation paths not easy. In addition, the user experience is also dependent on the background knowledge of an evaluator towards a question. In Question-1, for example, majority of evaluators have knowledge about the earthquake and hence prefer to browse opinion-related clips only. Factoid clips appear to “distracting” them because the provided captions do not clearly describe the content of video. They have to spend time to browse some of the clips before switching to another navigation path. In contrast, for Question-3, all the evaluators prefer summary more than opinion clips for no knowledge of traditional architecture in British. While the use of different colors have distinguished

4http://philogb.github.io/jit/
Figure 6.4: Precision of sentiment-oriented clip retrieval on the four opinion questions in Table 6.1.

between opinion and factoid clips, evaluators commented that, with limited familiarity to the question, it is actually not easy for them to choose the start point for navigation.

6.3 Summary

We have presented Constellation for exploratory browsing of video clips based on integration of different pieces of works presented in the previous few chapters. Particularly, different types of links are created and colored to connect videos and hint the perceived signals underlying them. In terms of user engagement and experience, the user studies clearly show the advantage of Constellation over list-wise presentation. The current interface suffers from several practical problems, such as the display of constellation become crowded with dense graph and the absence of dynamic update of constellation based on user knowledge, which will be issues worth further research.
CHAPTER 7

CONCLUSION AND FUTURE DIRECTIONS

In this thesis, several novel methods for analyzing the perceived signals (i.e., face, sentiment and emotion) in multimedia content have been presented and an exploratory search browser is also presented by combing these methods. This chapter briefly recapitulates the main contributions of this thesis and discusses several promising future directions.

7.1 Summary of Contributions

In this thesis, we have made contributions to unconstrained visual content understanding by proposing novel approaches to analyzing the key perceived signals – face, sentiment and emotion. The proposed exploratory browser for large scale video collection based on these three signals also demonstrates the practicability in real-world application. As far as we know, several problems in this thesis is the first time to be studied, for example, between-video celebrity naming in Section 3.2, multimedia opinion-oriented QA in Chapter 4 and the emotion-based cross-modal retrieval in Section 5.5. The major contributions of our works are summarized as follows:

• **Within-video naming with spatial-temporal relationships.** We have presented an effective CRF-based model (Section 3.1) for naming the faces of unconstrained Web video with the names in corresponding metadata. In the scenario of incomplete and noisy metadata, making better use of the limited information is critical for improving the performance. Specifically, CRF softly encodes the knowledge from Web images, the appearance of faces under different spatial-temporal contexts and their visual similarities to characterize the sets of faces and names. In addition, the “null” assignment problem is also elegantly tackled by considering the uncertainty in modeling unary and pairwise potentials.
• **Between-video naming with social relationships.** To leverage more clues for constructing relationships in CRF, we further extend the within-video face naming to between-video naming (Section 3.2). After the extension, more effective messages are passed to the isolate faces in within-video model and these faces have a high chance to be correctly named. We extensively verify the merit of the proposed CRF-based face naming on a large archive of Web videos, which leading to higher accuracy in face labeling but with minor degradation in speed efficiency.

• **Opinion clip localization.** We have devised a novel EM learning method (Section 4.1 and Section 4.2) to analyze the multimodal information, including visual, auditory and textual modalities, for locating the segments likely expressing opinions. Here, the opinion clips are limited to those showing opinion holders since only in such clips the advantage of using videos as answers to opinion-oriented clips can be demonstrated. Our empirical experiment on eight topics have justified the effectiveness of the proposed method on localizing the opinion clips.

• **Opinion question answering by deep learning.** We tackle short text matching problem with a deep architecture (Section 4.3) which mines the nonlinear relationship between text words in questions and speeches in videos. The proposed method is trained on the large scale QA archives in Yahoo!Answer and the latent semantics between question and answer are learnt, which is potentially more powerful to bridge the lexical and stylistic gaps between question and answer. The empirical experiments shows clear advantages of our proposed method in question-answer matching than the state-of-the-art methods.

• **Affective analysis and retrieval by DBM learning.** In Chapter 5, we presents the unsupervised learning of joint representation over multiple modalities with DBM. This representation captures the non-linear and complex correlations among different modalities in a high-level semantic space. The learnt joint representation is also much more compact comparing to the hand-crafted features. The most important advantage of the proposed method lies in its ability to generate the joint representation even with missing modalities. Hence, the proposed model naturally supports cross-modal retrieval beyond emotion prediction. The extensive experiments on several
public datasets have justified the effectiveness of the joint representation in both emotion prediction and retrieval.

- **Video browsing with “Constellation” in mind.** We have integrated the proposed works into a system for better understanding and exploration of multimedia content (Chapter 6). Our main contribution is on showcasing the feasibility of building up such system, by creating different kinds of hyperlinks based on various signals, for visualizing different aspects underlying the video clips. The visualization is beyond flat display of video list, where users can navigate the stories along the hyperlinks. The user study on TRECVID LNK dataset demonstrates the advantage of Constellation over list-wise presentation of video clips.

## 7.2 Future Directions

This thesis has laid a foundation for analyzing the perceived signals in unconstrained videos and it opens up several avenues for future works in unconstrained visual content understanding and browsing.

- **More effective facial feature representations.** While the overall performance of the proposed face naming approach is encouraging, the effectiveness is still limited by facial feature similarity, which is used in the unary energy term and pairwise visual relationship. The need for a sophisticated facial feature representation is expected to be even crucial for non-celebrities because there might not have enough Web images available for training face models. With the recent advancement in facial feature representation such as DeepFace [35] and face track [120], one direction of future work is to investigate the effectiveness of incorporating these representations into the proposed CRF framework. The main difficulty for training DeepFace lies in the need to collect large amount of training data with face labels, where such data are not publicly available.

- **Opinion question answering generalizaiton.** This thesis considers only topic wise learning of deep architecture for question-answering. Although attempts about generalizing the opinion QA system have been made in the proposed constellation browser, the deep architecture actually is not changed and only increase the coverage of training data. Recently, word2vec [19] has
been widely used for representing both syntactic and semantic meanings. Some works \cite{121} have been conducted for topic modeling based on the word vectors. Hence, one direction of future work is to take advantage of the deep topic modeling for the input feature generation. Furthermore, we can combine the topic modeling and QA model learning into an unified architecture to finetune the parameters accordingly and directly from raw text input to the output matching score.

- **Opinion clips analytics.** A ranked list of sentimental clips is not intuitive for exploring the opinions. One of the easiest way is to organize the opinions in temporal order to present the evolving public opinions towards specific events or objects. However, the sources for generating the temporal information of opinions, such as uploading date, transcript and captions, are not always accurate. Hence, instead of sorting the opinions in fine granularity, such as date, a better solution is to group the opinions into coarse granularity, such as week, month or year, and extract representative opinions to help users quickly grasp the gist of evolving public opinions.

- **Temporal context for joint feature learning.** In videos, the temporal information provides important context for understanding visual content. However, this thesis did not make use of this kind of information and the videos are represented by the average pool on all the keyframes. Recently, 3D convolutional neural network (3D CNN) \cite{122}, which extends the CNN with temporal information, has achieved success in many fields, such as action recognition. Thus, including the studies of temporal information into the multimodal DBM learning is a good direction to boost the effectiveness of emotion prediction.

- **Emotional concept description.** The advent of video sharing sites and rapid development of video technologies have led to explosion of video content. Providing emotion concepts description is in high demand for indexing since emotional videos are the most popular ones. Although our DBM model can provide textual description about the emotional concepts appeared in the visual content, these are just keywords list with ambiguous combination of nouns and adjectives. In addition, these keywords are given on the video level. Hence, there is a need for sequential recounting on different shots of the video. The Recurrent Neural Network (RNN) \cite{123} has shown its effectiveness in sentence generation. One of the future direction is to unify these
two networks to describe videos with emotion sentences.
REFERENCES


LIST OF PUBLICATIONS

Journal Publication


Conference Publication


• W. Zhang, **L. Pang** and C. W. Ngo, FashionAsk: Pushing Community Answers to Your Fingertips, ACM Multimedia (ACM MM), Nara, Japan, October 2012.

