MULTIMEDIA USER PROFILING IN ONLINE SOCIAL NETWORKS

GENG XUE

B.S., Northeastern University of China, 2012

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

SCHOOL OF COMPUTING

NATIONAL UNIVERSITY OF SINGAPORE

2017

Declaration

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

GENG Xue

February 15, 2017

Publications

- ***One of a Kind: User Profiling by Social Curation.*** Xue Geng, Hanwang Zhang, Zheng Song, Yang Yang, Huanbo Luan, Tat-Seng Chua, *ACM Multimedia 2014: 567-576.*
- ***Learning Image and User Features for Recommendation in Social Networks.*** Xue Geng, Hangwang Zhang, Jingwen Bian, Tat-Seng Chua, *IEEE ICCV 2015: 4274-4282.*

Abstract

Online Social Network Services (OSNs) have been evolving continuously while revolutionizing our lives over the past decade. They provide popular platforms to build social networks and enhance social relationships among people who share common interests, activities, backgrounds and real-life connections. Over the years, many types of OSNs have emerged, many of which are multimedia-based sites such as Pinterest, Flickr and Youtube. Furthermore, people have been sharing more and more multimedia contents over the years. However, the exponentially increasing media contents will make it difficult for service providers to tailor media contents to accommodate specific individuals.

To address the above issue, this thesis attempts to undertake the task of user profiling which is one of the fundamental tasks of personalization in OSNs. To the best of our knowledge, most existing approaches only focus on mining textual information to construct user profiles, while overlook the abundant shared media contents. Unfortunately, textual information may not provide complete and easy-to-grasp information to generate user profiles. Hence, this thesis, taking Pinterest as an example, focuses on developing effective and efficient approaches to model user profiles, by exploring rich user-generated multimedia contents including images, texts, together with domain knowledge.

The task of profiling users based on their rich media interactions in OSNs poses several great challenges. First, how to mine the extremely heterogenous and noisy media contents for user profiling; second, how to use domain knowledge to guide the media feature learning for human-understandable user profiles; third, how to use user-media interactions in OSNs to advance the task of modeling users; and fourth how to integrate domain knowledge and social collective intelligence together to obtain efficient and effective user profiles for personalized services. To address the above challenges, this thesis first introduces a data-driven user profile ontology and exploits the relationships

between concepts in the ontology to enhance media understanding for user profiling. The outcome is a human understandable user profile for efficient personalized services. The second part of this thesis presents a deep learning model to reveal the weak correlations of user-media connections for learning representative features of images and users simultaneously. The final part of this thesis describes a co-factorization approach to integrate the above multi-modal contents, domain knowledge and social user-media connections together into a framework to profile users in OSNs.

Extensive experiments conducted on large-scale real-world datasets demonstrated that our proposed models could yield significant gains in constructing effective user profiles based on the multimedia contents shared by users in online social networks.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This chapter first introduces the background of user profiling in OSNs with their distinguishable characteristics, and then highlights the motivation of user profiling in OSNs, followed by the challenges and solutions. Finally, the contributions of this thesis are briefly summarized.

1.1 Background

Online Social Network Services (OSNs), through which people can create, disseminate, and consume information, have evolved themselves while revolutionizing our lives over the past 15 years[1]. To date, a large variety of OSNs have thrived on the Internet, focusing on retail market (*e.g.*, Amazon), friendship (*e.g.*, Facebook), movie review (*e.g.*,IMDb), photo sharing (*e.g.*, Flickr), and so on. It is widely acknowledged that social media offers us valuable opportunities in both academia and industry [27].

Among them, many are multimedia-based sites, such as Pinterest[2] and Flickr. As this thesis take Pinterest as an example to verify the effectiveness of the proposed models, we introduce the Pinterest site in detail. Pinterest, which is

[1]The first online social network, FriendsReunited was launched in 1999, founded in Great Britain to reunite past school pals.

[2]`http://www.pinterest.com`

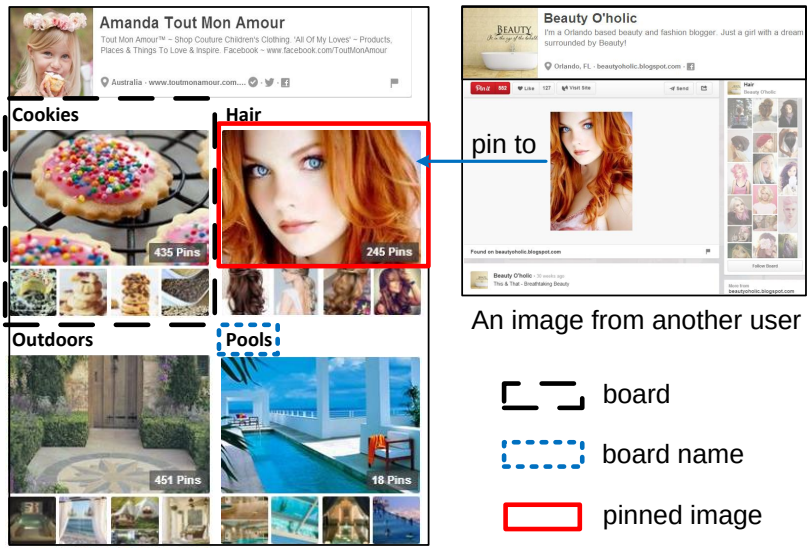


Figure 1.1: An example of "pin" in Pinterest. A user can curate image bundles called "boards". The curation is done by "pinning" images from other users. Here, the user pins a hairstyle image from another user into her own board named "hair".

the most popular SCS(Social Curation Service) registered by over 170 million [3] users, is a new emerging photo sharing social networking site that allows users to select, organize and keep track of images they like. Pinterest is a "pinboard-style" image sharing social network. The main innovation of it is to encourage users to collect and share interesting things in a categorized way. As illustrated in Figure 1.1, Pinterest innovates a notion called "*Pin to Board*", where users can '*pin*' or '*repin*' items they like into their own "*boards*". The key operation "pin" is to select a photo or video from external websites or another users' pin boards. The boards are bundles of pinned multimedia contents of various interest such as "Animals", "Arts", "Education" and "Fashion". For example, a user can have many bundles named "Cookies", "Outdoors", and "Pools" shown in Figure 1.1. In this way, social connections are encoded by pins, *e.g.*, users cannot directly send private or public messages to each other and the only social activity is to *like* a pin, *comment* on a pin or *repin* someone's pin into her own boards. Today, many conventional OSNs are inspired by this interesting feature of social curation, such as Flickr's "*add-to-gallery*".

Furthermore, more and more multi-modal data streams (*e.g.*, text, image,

[3] http://expandedramblings.com/index.php/pinterest-stats/

audio, video, *etc*) are generated as byproducts of people's everyday online activities in the digital world over the years. However, the exponentially growing media contents will make it difficult for service providers to offer interesting products to specific consumers. An effective and efficient user profile consisting of users' preferences on products will help to boost the performance of personalized services. Hence, it is essential to construct a comprehensive user profile in OSNs based on user media interactions. An effective and comprehensive user profile can advance many applications such as advertisement targeting, personalized recommendation, community detection and personalized web searching.

User profiling aims to establish user profiles by obtaining, extracting and representing the preferences of users [149]. User profiles can include demographic information, *e.g.*, name, age, country and education level [45]. A typical user profiling system comprises three intrinsic components: *information resources*, *user profiling* and *personalized services*. Figure 1.2 illustrates the framework of a typical user profiling system.

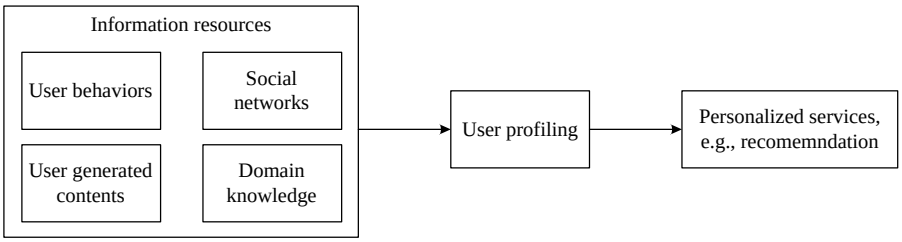


Figure 1.2: The flowchart of a typical user profiling system. This system starts from rich *information resources* mainly including behaviors, social networks, user generated contents and domain knowledge; the construction of user profiles is performed by *user profiling* based on the rich user data distributed in OSNs by which each user in the system is expressed as a representative profile. Finally, the learnt user profiles are applied to different *personalized services*, such as personalized recommendation.

1.1.1 Information Resources

User profile can be extracted from explicit or implicit sources. The *explicit* user profile is provided by users during the registration to some services, and

it is often incomplete and inaccurate. *Implicit* profiling is generally content-based, which has been shown to be a useful enhancement based on user relations (*e.g.*, followers or friends) and user-generated contents (*e.g.*, reviews and uploaded/shared photos, videos), which are often multimedia in nature [2]. Generally speaking, data resources that are used to construct user profiles are mainly split into four aspects: a) user demographic information; b) social networks; c) user generated contents (UGCs); and d) domain knowledge.

- *User behaviors* User behaviors such as browsing history [120] and query history [125] form as user implicit feedback to profile user preferences. For example, Sugiyama *et.al* [120] constructed user profiles based on modified collaborative filtering with detailed analysis of users browsing history in one day for personalized web search.
- *Social networks* To date, the information of social networks [91; 140] has been explored for modeling users based on the theory of *homophily* which states that people with similar interest tend to connect with each other and people of similar interest are more likely to be friends.
- *User generated contents* User generated contents can be *Keywords* (*e.g.*, tags), *Free Text* (*e.g.*, posts on Weibo, or tweets on Twitter), *Images* (*e.g.*, a user takes a snapshot and shares the photos on own social nets), *Videos*, and *Composite* of the above.
- *Domain knowledge* Domain knowledge is the information with a degree of certainty or community agreement. It provides a human-understandable, but machine-readable vocabulary describing a rich conceptualization of specific domain. Domain knowledge is an important element in understanding human behaviours. The *No Free Lunch theorem* [136] has implied that in order to gain in performance, a specialized algorithm that includes some prior human knowledge about the issue at hand must be used. The previous approaches [2; 147], however, only offer a general framework which is not perfect for a specific domain.

1.1.2 User Profiling

An efficient and effective user modeling approach in OSNs is required to handle the aforementioned different types of data resources. Traditional user profiling methods either employ feature engineers to generate hand-crafted meta-descriptors like fingerprint for a user or draw a set of latent features from a user's registered profile data, for example, through sparse coding [140]. Some approaches also use collaborative filtering techniques [5; 147] to infer user interests via collaboratively analysing group user behaviors, where the users are assumed to be independent with each other. However, most existing approaches only consider the textual information to profile users and ignore the user rich media interactions.

Furthermore, a comprehensive user profile often requires two important components: a latent-based user profile and a semantic-based user profile. The latent-based profile [5] is extracted by data-driven approaches such as the matrix factorization techniques that are able to somehow uncover the complex and unexpected patterns behind mass data. In contrast, the semantic-based user profile interprets the users in an understandable manner [2].

This thesis focuses on addressing the issue of user profiling based on their rich media interactions. To achieve this goal, we take into account several aspects including users, rich media, textual information and domain knowledge to model user preferences.

1.2 Motivation

With the tremendous development of OSNs, more and more multimedia data streams (*e.g.*, image, audio, video, *etc*) are generated as byproducts of people's everyday online activities in OSNs. For example, it has been reported that between April 2015 and November 2015, the amount of average daily video

"Spring Outfits & Trends 2016"

(a)

"Amazing Creativity With Nature - Amazing World"

(b)

Figure 1.3: An illustrative example of the role of rich media vs textual information. (a) the comments only indicates that the image is one piece of outfit without showing the contents such as "dress" and "bags"; and (b) the comments just tell us the picture is one piece of nature, overlooking the contents such as "sea" and specific designed "mountain".

views on Facebook doubled from 4 billion video views per day to 8 billion [4]. Recently visual contents have been considered very important in product marketing in almost every major social network, including Facebook, Twitter, Instagram and Pinterest [4]. Besides, many emerging multimedia-based sites such as Flickr, Pinterest and Snapchat have drawn more and more attentions. Hence, investigating user behaviors to infer users' diverse interests in these multimedia-based sites is urgently needed. However, to the best of our knowledge, most existing user modeling approaches only focus on mining the textual information in constructing user profiles [127; 2]. Unfortunately, textual information may not provide sufficiently complete and easy-to-grasp information to infer user profiles. Figure 1.3 shows two illustrative examples of the role of rich media vs textual information. We can observe that the comments of the images have not summarized the image contents accurately. Clearly, it will be much better if there are comprehensive analysis of rich media. Hence, this thesis focuses on modeling users based on user-media interactions and proposes to enhance media understanding and user interest understanding by incorporating multimedia content analysis, user-media connections and domain knowledge.

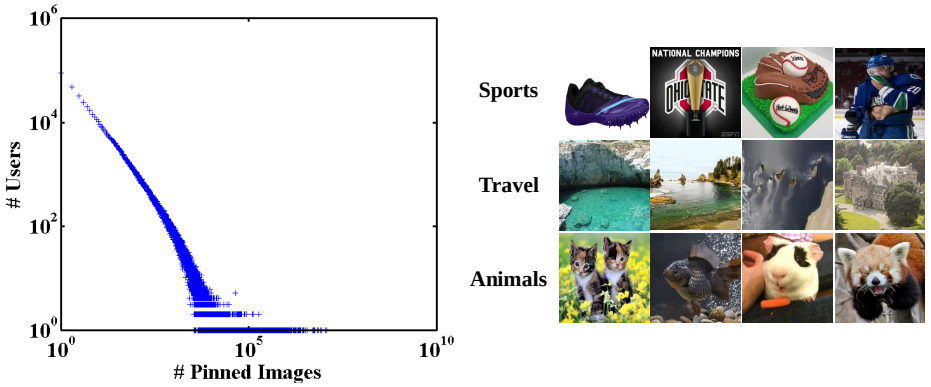
[4] http://blog.hubspot.com/marketing/visual-content-marketing-strategy

1.3 Challenges

It is worth mentioning that there exist several efforts dedicated to research on profiling users from rich media data. For example, [129] embedded deep content features into their model for music recommendation and Zhong [147] *et.al* have brought forward item features into their latent model for user profiling. However, they have not considered other important aspects such as domain knowledge, multi-modal contents and social connections in OSNs. To date, profiling uses based on rich media interactions is still an open issue in OSNs. There are mainly several challenges as follows:

- **Diverse and noisy media contents** To date, existing algorithms on media analysis is still limited [5]. When applied to OSNs, they may fail due to the diversity of OSNs, namely, extremely diverse and noisy multimedia contents. For examples, as shown in Figure 1.4(b), the contents of images in the same category are quite diverse. What's more, the noisy media contents comprise a large proportion. The extremely diverse and noisy multimedia would affect both the accuracy and efficiency of multimedia analysis for user profiling.
- **Heterogenous multi-modal contents** Moreover, online social networks (OSNs) are heterogeneous in nature where consumers share multi-modal contents with different modality expressing partial view of users interest [8]. For example, a user may share an image of an iphone with the comment of "Excellent phone with nice design!" to show his interest on the phone. This is another distinguishable feature of social media, namely multi-modality. Most existing approaches that analyze only one modality (*e.g.*, texts) will fail. Even some approaches [24] that attempt to mine multi-modality might fail since they purely analyze the multi-modal contents without considering the homogenous users.

[5] `https://www.ted.com/talks/fei_fei_li_how_we_re_teaching_computers_to_understand_pictures/transcript`



(a) (b)

Figure 1.4: Illustrations of extremely sparse user-content connections and diverse multimedia contents. (a) The power-law distribution of the number of images pinned by users. It means that the user-image connections are long-tailed and very sparse. (b) Some exemplar images of three interest categories. The contents of images in the same category are very diverse.

- **Sparse and noisy social connections** OSNs generally have two types of networks: friendship network among users and interest network between users and service items [140]. One of the fundamental mechanism that drives the dynamics of networks is the underlying social phenomenon of *homophily* [85]: people with similar interest tend to connect with each other and people of similar interest are likely to be friends. Modeling friendship network and interest network are equally challenging. This is because of the extreme sparsity of network structure in most OSNs [95]. Take Pinterest as an example, as shown in Figure 1.4(a), the frequency distributions of users and images follow the power-law distribution. In Pinterest, an ordinary user often curates around one hundred images which is only *one in a million* as compared to the whole Pinterest image collection. A social network is a large and sparse graph, involving hundreds of millions of users with each being connected to an extremely tiny proportion of the whole virtual world. Some traditional graph minging approaches may not be efficient to handle large scale and sparse friendship graph and interest-centric graph as well as reliably learning from rare, noisy and largely missing observations. This inspires us to

develop advanced approaches to combine the extremely sparse, and noisy social connections with content analysis to infer user preferences.

- **Machine understandable knowledge** The *No Free Lunch theorem* [136] has implied that in order to gain in performance, a specialized algorithm that includes some prior human knowledge about the issue at hand must be used. Knowledge is indispensable to understanding. An important question is, what is the meaning of the word "understanding"? Consider the following example. For human beings, when we see "24 Feb 1955", we recognize it as a date, although most of us do not know what it is about. Furthermore, if we are provided with a little more context, say "Steve Jobs, 24 Feb 1955, American", most of us would have guessed (correctly) the date represents Steve Jobs' birthday. We are able to do this because we possess certain knowledge, and in this case, "a date associated with a person might be his/her birthday". It turns out that what takes a human to understand the above example is nothing more than the knowledge about *concepts* (*e.g.*, persons, animals, *etc*.) and the intrinsic ability of relationships between different entities [138]. As introduced by Berners-Lee *et at.* [16] that the computer does not truly "understand" anything, but computers can manipulate content in ways that are useful and meaningful to the human consumers. This is the key point for knowledge understanding - we should not only let the machines provide the best answers but also understand them with explanation of how the answers were delivered. This triggers us to develop knowledge-guided approaches to enhance media understanding and user interest understanding.

1.4 Strategies

To bridge the aforementioned research gaps, the aim of this study is to design and develop a framework to infer multimedia user profiles based on user generated multimedia contents, user-media connections and domain knowledge. First, we propose to exploit human prior knowledge to guide the rich media understanding for user profiling. Second, we explore the information of extremely sparse social connections to learn a latent space for users and media simultaneously. Finally, we attempt to integrate the knowledge, social connections, rich media and textual information together to profile users.

1.4.1 User Profiling by Knowledge-based Multi-task Media Learning

At the beginning, we propose to exploit human prior knowledge to improve rich media understanding for user modeling since the knowledge can provide insight of domain concepts and relationships between them [127]. We first propose to automatically construct a data-driven profile ontology by pruning the Wikipedia ontology. Based on the fact that many visual cues are shared among sibling concepts in the ontology, we introduce a multi-task media learning approach [39] to advance media understanding for user profiles. Furthermore, we propose a low-rank algorithm to refine the user profiles by exploiting the various types of social cues including user-level, bundle-level and content-level. By conducting the above steps, we hope to learn the ontology-based user profiles that can be efficiently and effectively applied to the personalized recommendation application.

1.4.2 User Profiling by Deep Learning of User-Media Interactions

In this component, we exploit user-item connections in OSNs to enhance media understanding since user-item connections can reflect the valuable collective intelligence of user preferences on specific items. For example, if two images are shared by the same user, we may infer that the two images share some common visual properties. Meanwhile if two users share the same image, they may have some common interest. In this part, aside from rich media, we also aim to mine the heterogenous connections between users and images for user profiling. We present a novel deep learning framework that breaks down a large and sparse network topology into a tree-structured deep hierarchy. This deep model can compactly and efficiently learn representative features of users and images in a common low-dimensional space to reveal the weak correlations between images and users in the condition of the extremely sparse connections and extremely diverse images due to its deep structure. Besides, we propose a fast optimization algorithm that deploys an asynchronously parallel stochastic descent method based on the pow-law observation between users and items. This optimization algorithm can significantly reduce the time for the training of different user-image pairs.

1.4.3 User Profiling by Integrating Domain Knowledge, User-Media Interactions and Multi-modal Contents

In this component, we will integrate rich media, texts, user-media interactions and domain knowledge together in a framework to profile users. In particular, we attempt to learn the embedding of users, images and knowledge respectively by mining the heterogenous user-media associations and human prior knowledge i.e., color harmony and clothing ontology. Furthermore, the role of different data resources in the process of user profiling will be evaluated.

1.5 Research Contributions

This thesis mainly addresses the problem of profiling users based on rich media interactions in OSNs. Through exploring domain knowledge and social user-media connections, we propose to enhance media understanding and user interest understanding. Our main contributions stem from the proposed strategies of specific research problems. We summarize them as follows:

- We present a multi-task learning approach to build an ontology-based user profile. Different from the conventional semantic-based user profiling approaches, this framework is fully automatic and can be extended to general visual-oriented domain. Moreover, we explore the diverse multi-level social connections to refine the learned user profiles.
- We present a novel deep learning approach to learn the users and images into a low-dimensional space for fast and effective recommendation. Besides, based on the power law distribution between users and images, fast optimization algorithm that deploys an asynchronously parallel stochastic descent method is presented.
- We propose a framework to learn and integrate different aspects of social media contents including users, rich media, textual information and social connections into a common low-dimensional space. The learnt representations are able to support interpretable user profiles and fast image recommendation.

1.6 Organization

The remainder of this thesis is organized as follows. In Chapter 2, we offer a brief literature review of the broad domain of multimedia user profiling in social media. Chapter 3 discusses the technical details of the proposed ontology-based user profiling approach. In Chapter 4, we present a novel deep learning approach that maps the extreme user-image connections into a hierarchy. Chapter 5

focuses on a co-factorization approach based on rich media, textual information, user and domain knowledge. Chapter 6 concludes the thesis, highlights the limitations, and points to the future potential research directions.

Chapter 2

Literature Review

In this section, we will give a detailed survey on previous work which is related to our current and future work.

2.1 Multimedia Content Analysis

Currently, huge volumes of multimedia - images, videos, audio and texts are being generated and consumed in our daily life. Obviously, multimedia data is "big data" which offers us good chances to extract valuable information. It tells us about things happening in the world, topics of interest and gives clues about individual preferences [113].

However, different from previous research on structured and unstructured data, more effective algorithms for multimedia analysis are needed, which drives large amounts of research on "bridging the semantic gap" to enable large scale valuable information extraction [54].

2.1.1 Text Mining

Text mining deals with machine supported analysis of text [42]. It mainly uses the techniques from information retrieval (IR), information extraction (IE) as well as natural language processing (NLP). Current research tackles problems

of text representation [14], categorization [117], information extraction [104] and modeling of hidden patterns. The commonly used text features are strings (current commercial systems), single words (current statistical IR), named entities (IE systems) and linguistic units (NLP).

Text categorization The goal of text categorization is to classify documents into a fixed number of predefined categories [63]. Each document either belongs to exactly one or multiple categories. It has many applications to date, *e.g.*, assigning subject categories to documents to support text retrieval, routing and filtering. Many statistical and machine learning methods have been proposed including bayes probabilistic models [80; 66], factor analysis, nearest neighbor classification, decision tree [75], neural network [144], support vector machines [63] and combination of these with knowledge engineering. The main challenge of text categorization is the curse of dimensionality [61], since text features mostly use single word or some incorporate relations between words, *e.g.*, word-co-occurrence statistics, context information *etc*. Typical systems deal with 10 of thousands of terms. The "curse of dimensionality" obviously leads to more training data for most learning techniques.

Information extraction (IE) The goal of an information extraction system is to extract specific kind of information from a document [103], *e.g.*, web pages, medical notes, and news articles. For example, in the domain of terrorism, an IE system may extract the names of all physical targets, victims, and weapons in a terrorist attack. Since more and more text becomes available on-line, there is an urgent need for systems that extract information automatically from text data, especially free text. Besides, IE systems have been developed from structured text with tabular information to free text such as micro-blogs. The key point of IE systems is the text extraction rules that identify valuable information [114] which is different from the practical full-blown NLP systems which requires a complete analysis of document, IE system is a more focused and well-defined task.

Information retrieval (IR) The target of information retrieval (IR) of text is to find material (usually documents) of an unstructured nature (usually text) that satisfies an information need from large collections (usually stored in computers). In general, it often includes two stages: a) term selection and weighting for documents and queries; b) applying similarity measure to return top documents that satisfy users' needs. The IR techniques have been widely used to internet search engines, *e.g.*, Google, Bing and Baidu. Different search engines use various approaches to improve accuracies, such as Google uses the structures of links and Yahoo uses domain concepts. Current IR systems are still term-based. Further, Salton *et. al.* [105] have proposed a vector space model to represent query and documents.

Currently, free texts which are unstructured sequences of text with uncontrolled set of vocabulary has developed into the mainstream of real life communication and user generated short messages have been an important type of free texts. Many researchers have engaged themselves into free text processing. For example, classification of short text messages integrating other information sources such as Wikipedia [10] and WordNet [58]. Bharath Sriram *et. al* [117] proposed to use author information and features within tweets to classifies incoming tweets. Miles Efron *et. al* [37] proposed to use aggressive document expansion to improve information retrieval for short texts.

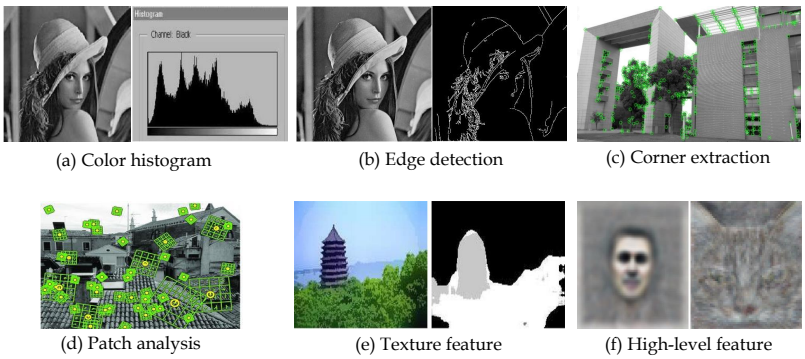
2.1.2 Image Content Analysis

The fact that large volume and variety of digital images currently acquired in different application domains has given rise to the requirement for efficient image management and retrieval techniques. Particularly, there is an increasing need for automated image content analysis and description techniques in order to retrieve images efficiently and effectively from large collections based on visual contents [141]. The extraction of image features is one of the fundamental techniques in image content analysis.

Feature Extraction

To date, there have been several kinds of features to represent the images: low-level, mid-level and high-level features which will be illustrated as follows.

- **low-level features** The low-level features [6; 29] is effective at capturing low-level image structure. It includes color, texture and shape features.



(a) Color histogram (b) Edge detection (c) Corner extraction

(d) Patch analysis (e) Texture feature (f) High-level feature

Figure 2.1: Image feature detection. (a) to (e) are low-level features while (f) is high-level features.

  - *Color features* Since color is quite intuitive, and simple, it's natural to identify an image using the color features. Color histogram [53] is the most commonly used approach to express the color features shown in Figure 2.1 (a). However, the main problem associated with the color feature is that the representation only relies on the the color of the objects appearing in the image, ignoring the shape and structure. Therefore, similar objects with different colors might be seen as different objects.
  - *Shape features* Depending on the applications, some require the shape representations to be invariant to translation, rotation, or scaling. The shape features of an object mainly include several approaches: 1) *edge detection* which aims to produce a *line drawing*, like the face in Figure 2.1 (b), something akin to a caricaturist's sketch; 2) *corner detection* which can be seen as detecting points where lines bend very sharply with high curvature, as shown in Figure 2.1 (c); 3) *patched/region analysis* which are the more

modern approaches to detect the localized patches of interest. For example, the more modern approach SIFT features [78] which transforms the images into scale-invariant coordinates relative to local features as shown in Figure 2.1 (d).

  - *Texture features* The texture refers to the visual patterns that have properties of homogeneity, showing the innate property of virtually surfaces, such as clouds, tress and bricks. An image can be seen as a mosaic of different texture regions. To date, the texture analysis ranges from using random field models to multi-resolution filtering techniques such as the wavelet transform [79]. Here is an example of texture feature extraction shown in Figure 2.1 (e).
  - *Others*. Some researchers have engaged themselves into combining those features to improve the distinct representation of images. For example, Pass *et.al* [97] proposed a histogram-based method color coherence vector (CCV) incorporating spatial information. Gevers *et.al* [47] proposed to combine the color and shape invariants into a unified high-dimensional invariant feature set for object retrieval.

- **Mid-level features** The mid-level features are structured image descriptions [21]. Popular examples include spatial pyramids [71], bags of features [112] and higher-layer activations of convolutional neural networks [69]. The process of extracting mid-level features involves several modules such as coding, spatial pooling, normalization and nonlinear transformations.
- **High-level features**. The high-level features are class-specific feature detectors [72]. For example, Quoc V. Le *et. al* [72] proposed a deep structure using unlabeled images to extract high-level features to detect objects directly.

Applications of Image Analysis

Content-based image retrieval Content-based image retrieval (CBIR) is the application of computer vision techniques analysing the contents of images to the image retrieval problem. Many content-based image retrieval systems can be described by the framework shown in Figure 2.2. The process includes extracting distinct features of images, building index, matching and visualizing result images.

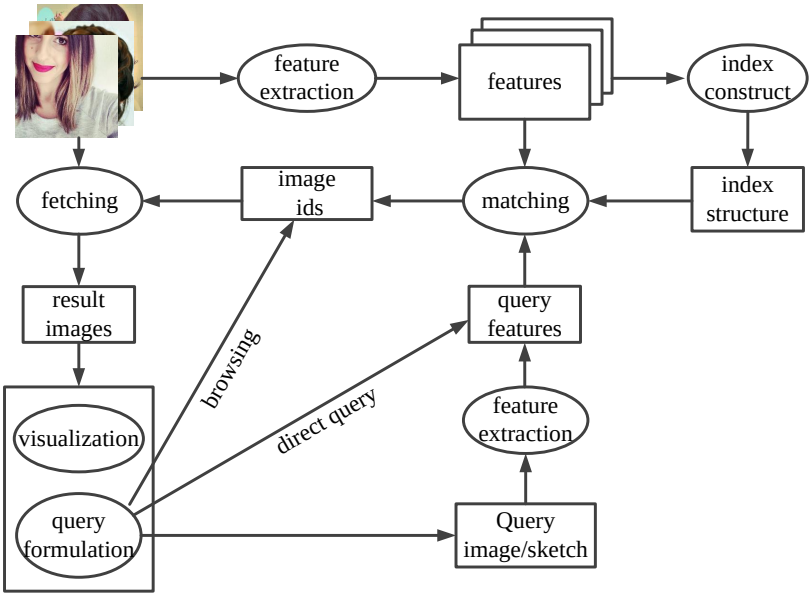


Figure 2.2: Content-based image retrieval framework.

2.1.3 Video Content Analysis

The advanced techniques in data capturing, storage, and communication have made large amounts of video data available to consumers. However, currently, we still have limited tools to describe, organize and manage video data. It is quite time consuming - and thus more costly - to generate content description. The core research in video content analysis is to automatically parse video, audio, and text to identify meaningful structures and extract, represent content attributes of video sources [34]. Different applications of video content analysis include event detection, motion detection, shape recognition, object

detection, video tracking *etc*. A typical scheme of video content analysis and

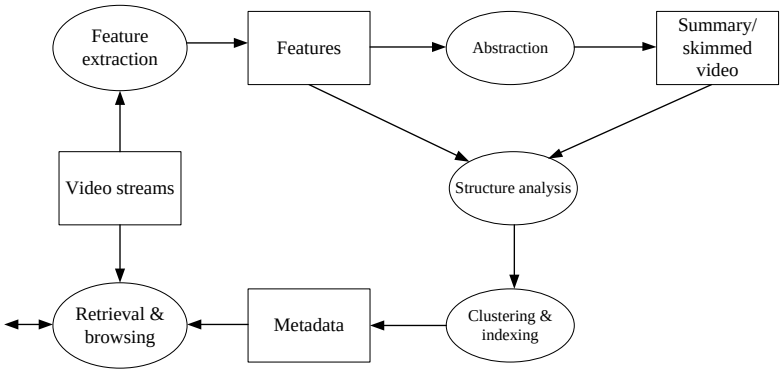


Figure 2.3: The primary analysis in video content analysis.

indexing involves four primary processes: feature extraction, structure analysis, abstraction and indexing as shown in Figure 2.3. Each process poses many challenging research problems.

- **Feature extraction** There are mainly two kinds of low-level visual features: a) static features such as GIST [96], SIFT, and colorSIFT [128]; b) dynamic motion features such as Space-Time Interest Points (STIP) [70], dense trajectory features (DTF) [134] and MoSIFT [25].
- **Video structure analysis** Video structure analysis lets us to manage video data according to temporal structures and relations and thus build table of contents. Many effective and robust algorithms for video parsing have been proposed for dividing videos into individual scenes [142].
- **Video abstraction** Video abstraction is the process of creating a brief representation of visual information about the structure of a video, which is much shorter than the original video. In this process, we need to extract a subset of video data from the original video such as key frames as entries for shots and scenes. Moreover, key frames which are still images extracted from original videos, play a significant role in the video abstraction process.
- **Indexing for retrieval and browsing**. Based on the above process whose results are often referred as the meta data of videos, we need schemes and tools to exploit these content meta data to query, search and browse large

scale video datasets.

2.1.4 Multimedia Content Analysis

Over the past decade, there have been an explosive growth in the amount of available multimedia information in our daily lives as shown in Figure 2.4. The internet is giving a vast mass of multimedia information repository. At the same time, digital cameras and recorders are becoming more and more popular with the result that the content of multimedia are expanding at an exponential speed. Almost all the personal computers and digital terminals store digital images and video content, and more new content is being created in every second. The demands from people for visual media content (image and video) is becoming more varied and broad. A wide range of digital devices including personal computers, digital televisions, cell phones and tablets will be able to access to images, video and other information plays an important role for the enrichment of people's life, work, education, entertainment and so on. People need much wider range of multimedia content. This trend necessitates the research and development of content-based multimedia analysis, understanding, filtering, monitoring and surveillance techniques. The ability to analyse, index and retrieve such multimedia contents, especially as they are being produced in real-time, will be of paramount importance.

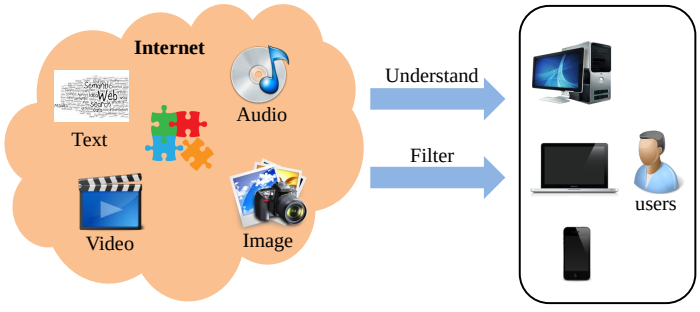


Figure 2.4: An exponential multimedia growth.

However, with a huge volume of online shared multimedia contents, how to mine the multimedia contents and further combine multi-modal contents in

different applications still remain an open issue. During the past several years, a large and growing body of literature has investigated feature learning of different modalities, especially the smart Convolutional Neural Network (CNN) [69] in the deep learning community which significantly improves the performance of image feature extraction. Additionally, Tang *et.al* [123] have proposed a semi-supervised representation learning method for text data. The above methods, however, focus only on uni-modality cases, which may not accomplish the task in a comprehensive way. To address this issue, different approaches [24] that aim to learn multi-modal representation have been proposed. For instance, Chang [24] *et.al* developed a multi-resolution deep embedding approach to learn a network of scalable, dynamic and heterogeneous data into a low-dimensional feature space.

2.1.5 Multimedia Data Analysis in OSNs

Social media services such as Facebook, Youtube, and Flickr and other web sites provide opportunities for people to share multimedia contents in an immense scale. For instance, Flickr users have shared over 4 billion images and videos on the site as of November 2009 [1] and Facebook users share a similar number of photos per month [3]. In 2015, Youtube users upload 400 new hours of video content per minute[1]. Such multimedia information might include many aspects: textual descriptors, location information of the content captured, the camera meta data, user information and social network context. The extra meta data from social network itself can advance and augment multimedia content analysis. Moreover, explicit user input tags and comments [30] as well as implicit references from users such as click streams can also be used to support multimedia content analysis in online social networks.

Multimedia content analysis is still a quite difficult problem as mentioned above. Meanwhile, the characteristics of online social networks make it

[1] `http://tubularinsights.com/hours-minute-uploaded-youtube/`

more difficult to analyse multimedia contents considering its own limits and challenges. For instance, the aforementioned contextual data and available meta data are noisy and inaccurate, sometimes misleading which leads to very little "ground truth" for online social network applications. Besides, the noisy and lack of semantics make the user provided meta data such as tags difficult to use. For example, an image tagged by "The King of Cat" may appear to be a lion, cat or Elvis Presley (a singer).

Most importantly, there is a shift focus of social multimedia analysis from that in traditional multimedia applications. First, it does not require general detection or classification tasks, *e.g.*, recognizing a "tiger" or a "cat". In contrast, tasks are narrower and more complex, *e.g.*, identifying a concert of a certain singer launched in last month. Second, it focuses more on precision, diversity and effective presentation instead of retrieving *all* relevant social media resources [94]. Third, the scale of social multimedia data is evolving all the time. In particular, the visual nature of the web has increased exponentially in recent years[2] while previous data gathering mainly comes from text or social connections. This phenomenon may require the development of more efficient and effective algorithms to integrate multiple media streams data and characteristics of online social networks to better support social multimedia applications.

Moreover, for personalized social multimedia applications, we need to link the diverse users with these shared multimedia contents. The heterogenous networks of users and contents would make the task of personalized services more difficult. Recently, many studies inspired from the notion of collective intelligence have been introduced [111; 146]. One of the most popular approaches, collective matrix factorization [111], has been widely employed to simultaneously factor several matrices, sharing parameters among factors when an entity participates in multiple relations.

[2] http://www.kpcb.com/insights/2013-internet-trends

2.2 Deep Feature Learning for Media

The performance of machine learning methods heavily depends on data representation (or features) to which they are applied. For this reason, a large amount of effort goes into the design of the data preprocessing and transforming which results in a distinct representation of data that supports effective and efficient machine learning methods [13]. Good representations are *expressive*, namely, a reasonable representation would capture a huge number of possible input configurations. A simple approach to evaluate the expressiveness of a model generating representations is on how many parameters this model requires as compared to the number of configurations it is able to distinguish. Traditional representation learning methods such as traditional clustering approaches, Gaussian Mixtures [46], Nearest neighbourhood algorithms, Decision trees, or SVMs, all require $O(N)$ parameters to distinguish $O(N)$ input configurations. However, modern deep learning methods such as restricted boltzman machine (RBM) [56], auto-encoders, can represent up to $O\left(2^k\right)$ input configurations using only $O(N)$ parameters. These are all distributed or sparse representations. The rapid increase in scientific activities has been nourished by a series of successes both in academia and industry. Here, we show several significant models in deep learning in detail.

2.2.1 Restricted Boltzman Machines

Restricted Blotzman Machines (RBMs) have been used as generative models for many different types of data including labeled or unlabeled images [56], bag of words representing documents [118] *etc*. The RBMs is a two-layer neural network which can model a training set of binary vectors. A graphical depiction of an RBM is shown in Figure 2.5. The energy function $E(v, h)$ of an RBM is defined as:

$$E(v, h) = -b'v - c'h - h'Wv \tag{2.1}$$

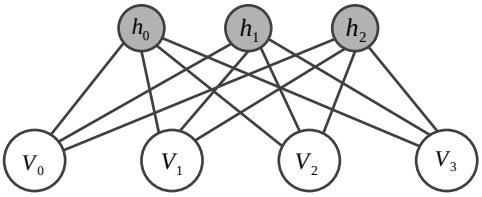


Figure 2.5: Restricted boltzman machine.

where W represents the weights connecting the hidden and visible units and b, c are the offsets of the visible and hidden layers respectively. This translates directly to the following free energy formula:

$$F(v) = -b'v - \sum_i log \sum_{h_i} e^{h_i(c_i + W_i v)} \tag{2.2}$$

which can be used to represent the likelihood based on the following formula:

$$P(x) = \frac{e^{-F(x)}}{Z}, Z = \sum_x e^{-F(x)}. \tag{2.3}$$

In the end, maximum likelihood are applied to this to update the parameter b, c, W.

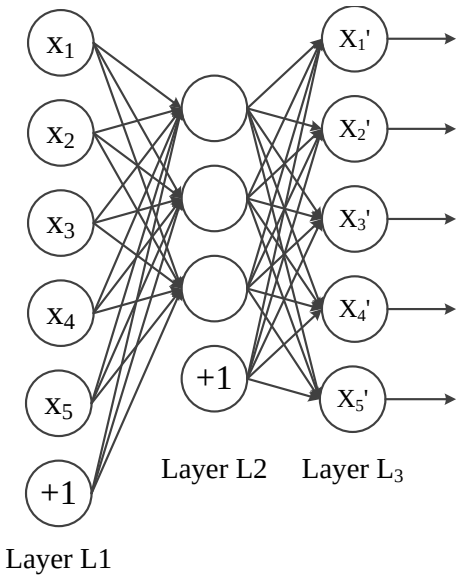


Figure 2.6: A simple example of autoencoder.

2.2.2 Sparse Autoencoder

Given a set of unlabeled training examples set, an autoencoder neural network is to extract distinct features to achieve the target that the last activation value is equal to the inputs via forward propagation and back propagation in an unsupervised way. Figure 2.6 shows a simple example of autoencoder aiming to learn a function that makes the output x' equal to the input x. Meanwhile, a sparsity constraint has been imposed on the hidden units in general case to discover interesting structures in the data since it is sufficient to obtain good generalization when the total number of bits to encode the *whole training set* is small as compared with the size of training set [12].

2.2.3 Convolutional Neural Network

The convolutional neural network architecture [55; 69; 110; 121] has been widely used in different kinds of applications. Specially, it has quite good performance in computer vision area. For example, Krizhevsky *et. al* [69] using the convolutional neural network has achieved the ImageNet classification benchmark. Table 2.2 summarizes the literatures on image feature extraction including several benchmark convolutional neural network models.

The very important part of convolutional neural network is convolution. Convolution of a $N \times N$ image using a $K \times K$ kernel can be understood as sliding a $K \times K$ window over the input image iteratively. For each position of the next layer, the value is equal to the dot product (sum of the multiplication of the corresponding pixels) of the kernel with the input pixels lying in the previous layer. In Figure 2.7, we have shown the calculation of the first two values of the second layers, where the convolution is implemented by a 6×6 image with a 2×2 kernel W.

The convolutional neural network integrate three architectural ideas ensuring shift and distortion invariance to some degree: local receptive fields, shared weights, and sometimes, spatial or temporal sub-sampling [73]. The local

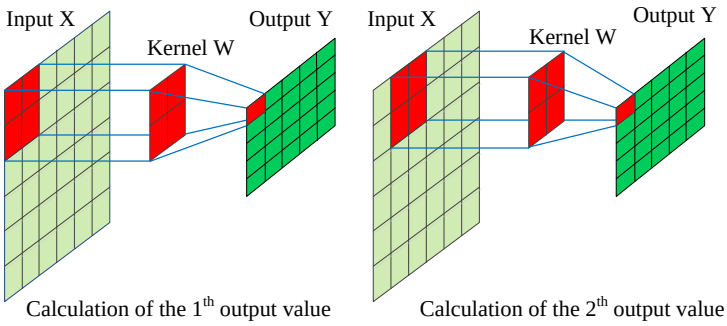


Figure 2.7: Description of 2D convolution.

receptive fields shared the weight (kernel W in Figure 2.7) at different places of images to extract elementary features such as oriented edges and corners. Then those output features are combined by the high layers. Besides, each convolutional layer often has several feature maps (with different weight vectors), so that multiple distinct features can be extracted at each location.

Figure 2.8 is a typical convolutional neural network architecture proposed by Krizhevsky *et. al* [69]. From the figure, we can easily see that the input of this architecture is the raw RGB pixel intensity values of a 224×224 image. These values are forward propagated through 5 convolutional layers with pooling and non-linearities along the way and three fully connected layers to determine its final neuron activation: a distribution of over 1000 object categories.

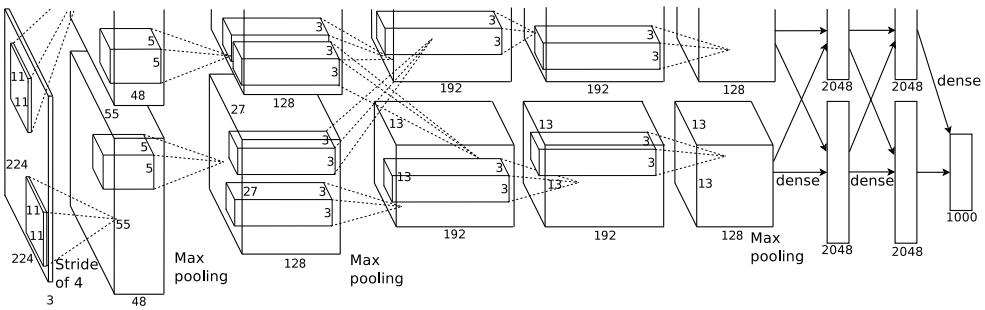


Figure 2.8: A typical convolutional network structure in training Imagenet [69].

There are also other state-of-the-art deep neural network, such as deep belief network [56] where RBMs are stacked and trained in a greedy manner, de-noising autoencoders and recurrent neural networks [14].

2.2.4 Recurrent Neural Networks

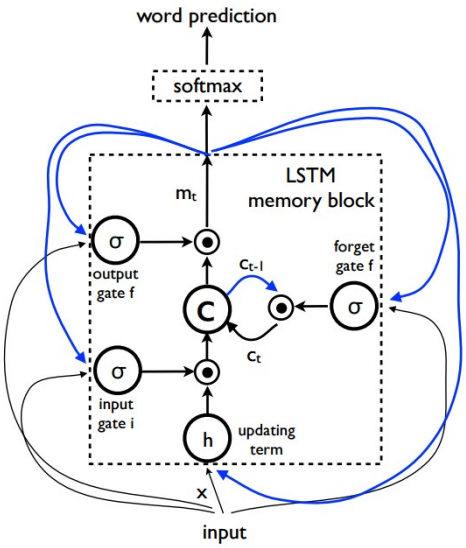


Figure 2.9: LSTM: the memory block contains a cell c which is controlled by three gates. In blue we show the recurrent connections the output m at time $t-1$ is fed back to the memory at time t via the three gates; the cell value is fed back via the forget gate; the predicted word at time $t-1$ is fed back in addition to the memory output m at time t into the Softmax for word prediction [131].

The Recurrent Neural Network (RNN) is neural sequence model that achieves state of the art performance on many tasks that include language modeling [89], speech recognition [50], and machine translation [64].

Figure 2.9 shows a particular form of recurrent neural nets, LSTM. LSTM is introduced to to deal with vanishing and exploding gradients [57], the most common challenge in designing and training RNNs.

2.2.5 Challenges of Deep Architectures

Table 2.1 concludes the characteristics of different deep learning architectures. Deep architectures have not been discussed much because of its poor training and generalization errors using standard random initializations of parameters [12]. Many experimental results have shown that gradient-based training of deep supervised multi-layer neural networks often get stuck in "local minima". Besides, when the architecture becomes deeper, it becomes more difficult to obtain good representations. Moreover, insufficient depth

Table 2.1: The summarization of classical deep learning architectures.

Methods	**Type**	Ideas	**Applications**
Convolutional neural network [69]	Supervised	Instead of learning single global weight matrix between two consecutive layers, it aims to find a set of locally connected neurons.	Image recognition
Recurrent Neural Network [57]	Supervised	It performs the same task for every element of a sequence, with the output being depended on the previous computations.	Language modeling, speech recognition and machine translation
Sparse Autoencoder [130]	Unsupervised	When you pass data through such a network, it first compresses (encodes) input vector to "fit" in a smaller representation, and then tries to reconstruct (decode) it back. The task of training is to minimize an error or reconstruction	Dimensionality reduction.
Restricted Boltzman Machines [56]	Unsupervised	It shares similar idea with auto-encoder approaches. However, instead of deterministic (e.g. logistic or ReLU) it uses stochastic units with particular (usually binary of Gaussian) distribution.	Dimensionality reduction, classification

Table 2.2: The summarization of related works on image feature extraction.

Related Works	**Methods**	**Type**	**ILSVRC-2012 Top-5 Error**
[Lowe 2004]	SIFT	Hand-crafted feature	26.7%
[Dalal 2005]	HOG	Hand-crafted feature	
[Ahonen 2006]	LBP	Hand-crafted feature	
[Krizhevsky 2012]	AlexNet_CNN	Neural network feature	15.3%
[Simonyan 2014]	VGGNet_CNN	Neural network feature	7.32%
[Szegedy.C 2014]	GoogleNet_CNN	Neural network feature	6.66%
[He 2016]	ResNet_CNN	Neural network feature	3.57%

of architectures can hurt. However, many applications can be represented efficiently with deep architectures and cannot be represented efficiently with shallow architectures [12]. This indicates that the design of deep architecture plays an important role to good generalization.

2.3 User Profiling

The emergence of Word Wide Web, smart mobile devices and online social networks have revolutionized the way we communicate, create, disseminate, and consume information. However, such large scale of the web is limiting its use since there is a sea of internet information, the consumers have to do all the work to use the web. For example, search engines often provide the same results for different preferences, intentions and contexts without considering the specific needs of users; they expect users to spend additional efforts to accomplish their searches. Therefore, personalization can be the solution that customizes web contents to the needs of specific users, taking advantage of the information

from the analysis of the user's behaviours [38]. In particular, one of the basic components of personalization is user profiling.

User profiling aims to establish user profiles by obtaining, extracting and representing the preferences of users [149]. A user profile might include demographic information [40; 115] such as name, age, country and education level. Sometimes it can also represents the interests or preferences of either a community of users or an individual [45]. As shown in Figure 1.2, in general, the user profiling process mainly includes three main phases [45]: a) an information data collection process which gathers rich data; b) a user modeling approach to mine useful information from rich data resources to construct and express user profiles; and c) a personalized service that integrates the learnt user profiles in OSNs. Table 2.3 has summarized recent literatures on user profiling.

2.3.1 Information Resources

Generally speaking, the data resources that are used to model users are split into four aspects: a) user demographic information; b) social networks; c) user generated contents (UGC); and d) domain knowledge.

- *User behaviors* User behaviors such as browsing history [120] and query history [125] form as user implicit feedback to profile user preferences. For example, Sugiyama *et.al* [120] constructed user profiles based on modified collaborative filtering with detailed analysis of users browsing history in one day for personalized web search.
- *Social networks* To date, the information of social networks [101; 102] has been explored for modeling users due to the theory of *homophily* that people with similar interest tend to connect with each other and people of similar interest are more likely to be friends. For instance, Mislove *et.al* [91] use friendships to infer Facebook users' attributes. They developed a clustering algorithm to find communities in the network and then assigned an identical attribute value to users in the same

community. Yang *et.al* [140] presented a model to propagate interests of an item among users via their friendships.

- *User generated contents (UGCs)* From the perspective of user shared contents, user generated contents can be *Keywords* (*e.g.*, tags), *Free Text* (*e.g.*, posts on Weibo, or tweets on Twitter), *Image* (*e.g.*, a user takes a snapshot and shares the photos on own social nets), *Video*, and *Composite* of the above. To date, a large number of studies have been conducted to understand the contents and then obtain what the consumers are interested in [2; 5; 147]. For example, Abel *et.al* [2] have studied how to leverage Twitter messages posed by users for user modeling and evaluate the quality of user models in the context of recommending news articles.
- *Domain knowledge* Domain knowledge is the information with a degree of certainty or community agreement. It provides a human-understandable, but machine-readable vocabulary describing a rich conceptualization of specific domain. Hence, many researchers have conducted studies to construct knowledge-based user profiles for both good interpretation and efficient personalized services [44; 23; 127; 109; 86]. For instance, Trajkova *et.al* [127] used the Open Directory Project concept hierarchy (ODP, 2012) as their reference ontology to train different concept classifiers for constructing user profiles applied to web search. Sieg *et.al* [109] proposed to maintain and update user profiles as annotated specializations of a pre-existing reference domain ontology and presented a spreading activation algorithm for maintaining the interest scores in the user profile based on the user's ongoing behavior.

2.3.2 User Profiling

Traditional user profiling methods either employ feature engineers to generate hand-crafted meta-descriptors like fingerprint for a user or draw a set of latent features from a user's registered profile data, for example, through

Table 2.3: The summarization of related works on user profiling.

Related Works	Profiles	Information resources	Data sources	Data scale
[Sugiyama 2004]	search preferences	browsing history	Google search engine	50 query topics
[Teevan 2005]	user interests	queries, browsing history	MSN Search	15 participants evaluate the top 50 Web search results for approximately 10 self selected queries each.
[Middleton 2014]	research paper topics	Text, ontology	Research papers	260 subjects over an academic year
[Rao 2010]	age, gender, regional origin, political views	Text, Behavior, Relation	Twitter	1,000 users for gender, 2,000 users for age, 1,000 users for regional origin, 400 users for political views
[Bi 2013]	gender, age, religion, political views	Text	Facebook	457,000 users' Facebook data and 3.3 million users' search logs
[Querica 2012]	personality	Relation	Twitter	335 users
[Markoviki 2013]	personality	Text	Facebook	250 users (10,000 status)
[Bazelli 2013]	personality	Text	StackOverFlow	total posts on StackOverflow between Aug. 2008 - Aug. 2012

sparse coding [140]. Moreover, some approaches use collaborative filtering techniques [5; 147] to infer user interests via collaboratively analysing user behaviors, where the users are assumed to be independent with each other. However, most existing approaches only consider the textual information [81; 11; 19] to profile users and also have not taken the connections among users or user behavior information into consideration.

According to different user modeling approaches, the resultant user profiles usually are split into either latent-based user profiles [5; 147] or semantic-based user profiles [2; 52]. Seen from the angle of latent-based user profiles, Zhong *et.al* [147] have put forward a latent factor model purely on implicit negative and positive user feedback to infer user interest vectors. While from the point of view of semantic-based user profiles, Abel *et.al* [2] have built three types of profiles that differ with respect to the type of concepts: entity-, topic- and hashtag-based profiles for personalized news recommendations in Twitter. Moreover, Guy *et.al* [52] introduced a user vector of related people and tags for recommending social media items. However, to our best knowledge, very little research up to now considers both semantic-based profiles and latent-based profiles simultaneously.

2.4 Personalized Recommendation

Personalized recommendations involves a process of gathering and storing information about web site consumers, analysing current and past user interactive behaviors, and, based on the comprehensive analysis, delivering the user interested content to each consumer [26]. Traditional recommendations include three approaches: a) content-based recommendation method: this is the traditional content-based recommendation method [98], where users are recommended items similar to those they preferred in the past; b) collaborative filtering recommendation method: the user is recommended items that people with similar tastes and interests preferred in the past; c) hybrid recommendation method: these methods combine collaborative and content-based methods.

Let d be the function that measures the interestingness of item i to user u, *i.e.*, $d : I \times U \rightarrow R$, where R is a totally ordered set. Then for each user $u \in U$, we want to choose such items $i \in I$ that maximize the user's utility [4].

$$\forall u \in U, i'_u = \arg\max_{u \in U} d(i, u) \tag{2.4}$$

Each element of the user space U can be defined with a *profile* that includes various user characteristics, such as age, gender, income, *etc*. Similarly, each element of the item space I can represent a set of characteristics of the item. For example, in a movie recommendation application, each movie can be represented by its title, director, leading actors, *etc*.

2.4.1 Content-based Methods

Content-based filtering approaches recommend images based on a comparison between the contents of the images and a user profile [9; 107]. User profiles can be identified by the users themselves, or learned from the content of the images that users have rated. In CBF, the utility $d(i, u)$ of the item i for the user u is estimated based on the utilities $u(u, i')$ assigned by user u to items

$i' \in I$ that are "similar" to the item i. For instance, in a movie recommendation application, the content-based recommendation system attempt to explore the commonalities (*e.g.*, directors, specific actors, *etc*.) among the movies that the user u has rated highly in the past.

In content-based systems, the utility function $d(u, i)$ is usually defined as:

$$d(u, i) = score(ContentBasedProfile(u), Content(i)) \tag{2.5}$$

where $ContentBasedProfile(u)$ represents a vector of weights where each weight denotes the importance of a keyword to the user u and can be computed individually using a variety of approaches. And $Content(i)$ can be an item profile, which may includes a set of distinct attributes of item i.

However, content-based recommendation systems often have some limitations as follows:

- **Limited content analysis**. Content-based are often limited by the features with the automatically extracted feature techniques which might works well in text documents but not in other domains.
- **Over specialization**. Such content-based system only recommend users the items which are most similar to the items that users preferred in the past. However, in certain conditions, items should not be recommended if they are too similar [4].
- **Cold start**. For the user who have rated very few items will not have accurate recommendation results. That is, when a user only rates a limited number of images, the limited content information cannot be generalized to discover the user's broader interest.

2.4.2 Collaborative-based Methods

Different from content-based recommendation approaches, collaborative-based filtering (CBF) methods try to predict the utility of items for a specific user based

on the items preferred by other similar users [119]. The similarity between users are often computed based on the overlap of shared images. That is, the utility $d(i, u)$ is estimated using the utility $d(i, u')$ assigned to i by those users u' who are "similar" to the user u. Commonly, the value of the unknown rating $r_{i,u}$ for user u and the item i is usually computed as an aggregation of ratings from other similar (usually top N most similar) users for the same item i:

$$r_{i,u'} = \sum_{u' \in U'} r_{i,u'} \tag{2.6}$$

where U' is the top N user.

However, collaborative recommendation approaches also have several limitations as follows:

- **Cold start problem**. This is the same as with the content-based methods.
- **New item problem**. New items often be added regularly to the recommendation systems. Until the new item is rated by many users, the recommendation systems can be able to recommend it.
- **Sparsity problem**. In any recommendation system, the number of ratings is usually quite small compared with the number of ratings that need to be predicted.

To alleviate the sparsity problem, matrix factorization based CF models have been proposed, such as the singular value decomposition (SVD) [106], weighted matrix factorization (WMF) [59], and the combination of probabilistic matrix factorization (PMF) [92] and topic models [132]. These models assume that the user-image matrix has a low-rank reconstruction by low-dimensional user and image features. We argue that such methods are essentially "shallow" models since they directly seek the resultant high-level features from user-image matrix. When the matrix is very sparse, these methods will fail to find meaningful latent factors.

2.4.3 Hybrid Methods

Several recommendation systems attempt to use the hybrid approaches which integrate the content-based recommendation systems and collaborative filtering recommendation systems together. And it has different ways to combine the above two recommendation methods: a) implementing content-based and collaboratively filtering methods separately and combing their recommendations together; b) incorporating the content-based approaches into a collaborative approaches; c) incorporating the collaborative filtering methods into the content-based approaches; d) developing a unified model which can integrate the characteristics of the content-based and collaborative filtering methods.

2.5 Summary

As mentioned above, it is easily seen that it is still quite difficult to conduct multimedia content analysis, and the characteristics of the online social service (*e.g.*, sparse social connections, complex user behaviours) by no means makes it more difficult to analyse multimedia contents. Even we have more efficient state-of-the-art methods (*e.g.*, deep learning), we can not directly adapt those methods to such complex research problems. In addition, our target of mining effective and efficient user profiles in social media services, makes it essential that we should incorporate all facets of knowledge, range from individual information and expert knowledge (*e.g.*, Wikipedia), to extract valuable and machine understandable information which can be widely used to many applications such as personalized recommendation.

Chapter 3

User Profiling by Knowledge-based Multi-task Media Learning

In this chapter, we target at proposing a multi-task media learning approach for user profiling where relationships of siblings will be involved. In particular, we apply the proposed approach to infer users' interests for personalized recommendation. Extensive experiments have demonstrated the effectiveness of the proposed approach.

3.1 Introduction

As mentioned in 1.1.1, user profile can be extracted from the implicit resources such as user generated contents including images, videos and composite of them. It has been reported that the major interest of OSNs is rapidly shifting from text-based contents to multimedia[1]. Hence we propose to exploit user generated multimedia data to profile users.

Motivated by the promising outlook of social curation, we attempt to establish high-quality user profiles based on such new social media platforms, *i.e.*, SCSs (Social curation services), with the aim of advancing fundamental social applications such as recommendation. SCS is a new type of emerging social

[1] http://www.kpcb.com/insights/2013-internet-trends

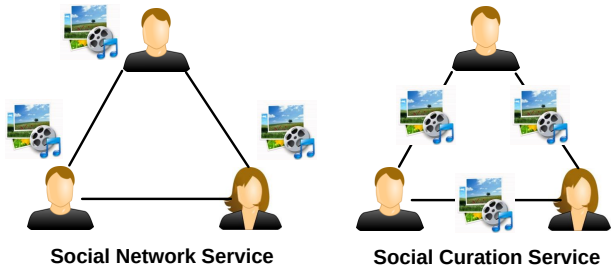


Figure 3.1: The illustration of the user-centric OSNs and the content-centric SCS. Although OSN contains user-generated content, the philosophy behind it is limited to user-user interactions. Alternatively, SCS encourages user interactions on the content level.

media platform, where users can select, organize andkeep track of multimedia contents they like. Specifically, our user profiling approach is superior by exploiting the distinguishable features of SCSs as compared to the traditional social network services for two key reasons:

Organized vs. Unorganized Contents. Unorganized multimedia contents in conventional OSNs are visually and semantically noisy and diverse, and thus are hard to be analyzed and exploited even with the state-of-the-art multimedia annotation techniques [60]. In contrast, SCSs contain a considerable amount of manually collected and maintained contents. For example, images in a curated bundle (*e.g.*, the board in Pinterest or the gallery in Flickr) are very focused on the same semantics as shown in Figure 1.1. Such organized multimedia contents offer us high-quality human labeled training data for multimedia modeling. Moreover, we are able to mine a large amount of curated bundles of user interest to build an content-based ontology to further structuralize the data, resulting in more personalized and accurate user profiles.

Content-centric vs. User-centric Network. As aforementioned, conventional user-centric OSNs are not optimized to create comprehensive user profiles based on user-generated contents. Alternatively, as illustrated in Figure 3.1, content-centric SCSs are advantageous in reliable social cues on user preferences. In particular, user curation through multimedia contents helps to encode multi-level content-content connections, which are expected to pinpoint the user preference in terms of the contents generated by the user. Such connections between

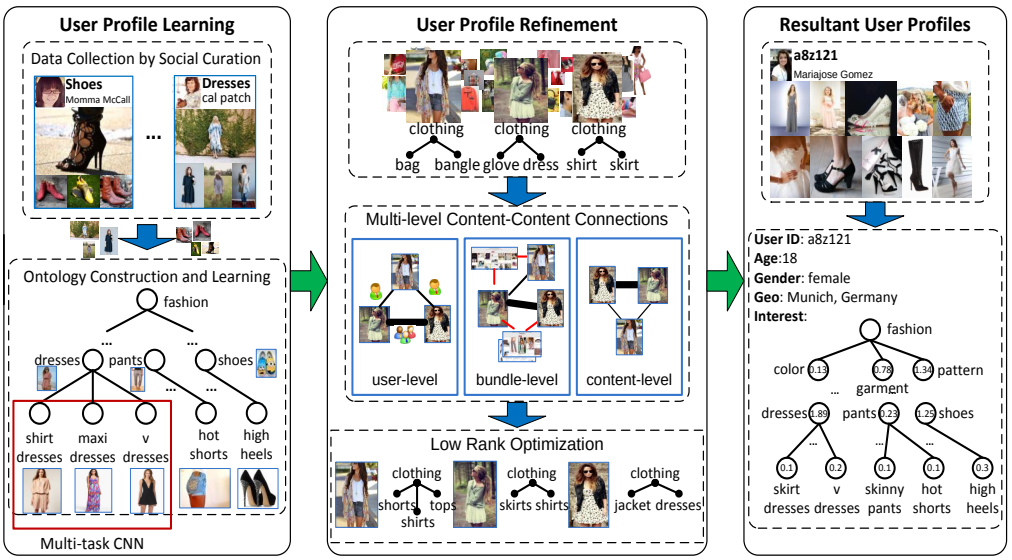


Figure 3.2: The overview of the proposed user profiling by social curation.

two images include: a) *user-level* connection, where the strength indicates how many users have pinned the two images, b) *bundle-level* connection, where the strength indicates how many bundles share the two images and c) *content-level* connection, where the strength suggests the similarities between the two images. In particular, the first two connections are expected to unravel the diverse user interest hidden in the contents. For example, if two images are only shared by few users (or bundles), the connection between them rarely suggests similar user interest. However, if they are shared by many users (or bundles), they tend to be very likely referring to the same interest. Our user profiling method can leverage rich information to refine the imperfect content-based profile models.

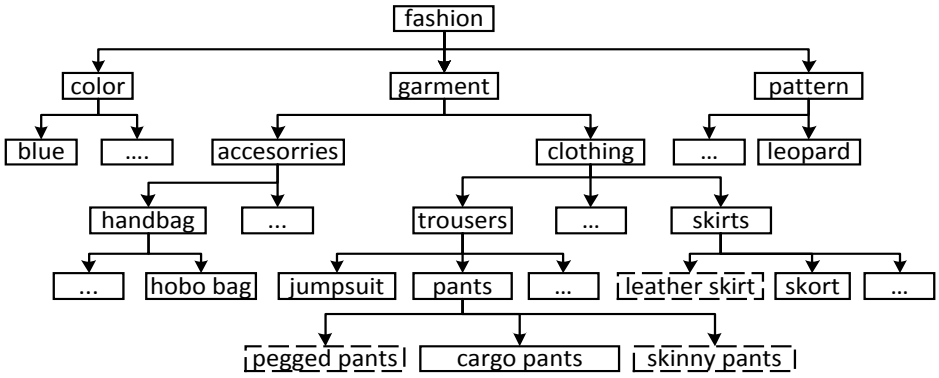
The overview of the proposed user profiling approach based on an example of SCSs, Pinterest, is illustrated in Figure 3.2. First, we collect multimedia contents curated by users, *i.e.*, the images in bundles as well as the associated user interest description like bundle names and tags. Due to user curation, the collected data are of high-quality and focused according to the user interest. Second, we propose an automatic ontology construction method to structuralize the curated images onto an ontology. The construction is done by pruning an expert ontology, *i.e.*, Wikipedia Category, to the desired user interest, *e.g.*, the fashion domain. Third, based on the constructed ontology, we are able

to learn content-based models to generate ontological user profiles, which are more comprehensive and personalized than the traditional text-based profiles. In particular, we propose a novel multi-task convolutional neural network (mtCNN) in order to leverage both the relatedness of sibling user-interest items and the cutting-edge advances in high-performance visual modeling. Forth, we further propose a low-rank recovery framework to further refine the generated user profiles by the ontological profile models, exploiting the rich user-level, bundle-level and content-level social relations offered by social curations. Therefore, the resultant user profiles are expected to retain: a) the interest of user, b) the interest of user-curated bundles and c) the semantic affinities with respect to the ontology, supporting effective fundamental social media applications such as recommendation. Experimental results on 1,239 users and 1.5 million images collected from Pinterest in fashion domain demonstrate that the proposed user profiling method is more effective than other state-of-the-art methods in terms of recommendation.

Our research is a pioneering work on content-based social curation analysis, with the following contributions:

- We propose a novel content-based user profiling method using social curation. Our work concentrates on exploring how social curation can help in content-based social multimedia analysis
- We present a user profiling framework on how to exploit the rich social information in SCS. This framework is fully automatic and can be extended to general visual-oriented domain. In this work, we use the fashion domain as an example.

The rest of the work is organized as follows. Section 3.2 describes the user profile learning process. Section 3.3 illustrates the process of user profile refinement. Experimental results and analysis are reported in Section 3.4, followed by conclusions in Section 3.5.



(a) Profile Ontology

#	Statistics
leaf nodes	427
total nodes	465
average leaf samples	546
least/largest leaf samples	177/11,442
least/largest depth	3/6

(b) Ontology Statistics

Figure 3.3: (a) The automatic constructed profile ontology in the fashion domain. The dashed boxes denote the automatically augmented items. (b) The ontology statistics based on user-generated contents collected from Pinterest.

3.2 User Profile Learning

Social curation service (SCS), by nature, contains high-quality images in bundles curated by users. Here, "high-quality" means the semantics of images are highly constrained and relevant by the user-provided tags. This gives us a great opportunity to develop well-generalized content-based models to predict the semantics of images, in our case, the user interest items.

3.2.1 Profile Ontology Construction

We use an ontology to organize the user interest items and their relationships in a domain from general to specific, as it has been widely shown to be effective in integrating human knowledge of the domain and data distributions to improve the modeling of visual semantics [143]. We propose to build an interest ontology to describe user profiles, for example, in the fashion domain.

After harvesting the user-curated interest items for pinned images such as comments and bundle names, we want to automatically generate a profile

ontology $\mathcal{O} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v\}$ is a set of items such as "leopard", "dresses" and an edge in $\mathcal{E}$ is an ordered pair of items in $\mathcal{V} \times \mathcal{V}$. Now, we introduce how to automatically construct the ontology $\mathcal{O}$ by mining the contents in Pinterest. First, we exploit Wikipedia Categories to build a preliminary ontology based on pre-defined general user interest. This ontology is rooted from three general nodes: "color", "garment" and "pattern", voted by the most general WordNet items which are hypernyms of the user provided fashion words. However, it is hard to adapt to the real user interest distribution of the user-curated data, since a) some items are outdated such as "polonaise", which are missing from the user-curated data and b) high-frequency items such as "V-dresses", "sleeveless dresses", on the other hand, are missing from this ontology. Therefore, we should prune the Wikipedia ontology to the user interest on demand.

Specifically, to remove the outdated items, we consider the items with low term frequency (*e.g.*, less than 100 times) derived from around 800,000 user-generated items as the outdated ones. Also, we need to add high-frequency items into this ontology. Note that this is not a trivial task since it is challenging to find which item node in the ontology is most semantically related to a given high-frequency item. Here, we propose a novel method for augmenting the ontology with out-of-vocabulary items. Suppose we want to add a high-frequency item h onto the existing ontology $\mathcal{O}$, the key is to find the most possible semantic path from top to bottom and then add h as a sibling of an item node v if v is most semantically similar to h among others along the path. In order to numerically calculate the most possible semantic path, we need to transform item words into numeric vectors. Here, we use Word2Vec [87] to transform an item into a 300-D vector, retaining its semantic meanings. Then, we use all the items in $\mathcal{V}$ to sparsely represent the h as

$$\arg\min_{\mathbf{a}} \|\mathbf{h} - \mathbf{V}\mathbf{a}\|_2^2 + \lambda\|\mathbf{a}\|_1, \tag{3.1}$$

where $\mathbf{h}$ is the 300-D vector of item h, $\mathbf{V}$ is a dictionary matrix which is arranged

by putting vectors of $\mathcal{V}$ column-wisely, $\mathbf{a}$ is a sparse coefficient vector and λ is a trade-off parameter. By doing this, each item node of the ontology $\mathcal{O}$ is assigned a value according to the sparse codes of h. Therefore, we can find the most possible semantic path whose nodes have the largest sum of sparse codes. Finally, we can find the target item node v along the path which has the largest sparse code value. As a result, we obtain a comprehensive user profile ontology as shown in Figure 3.3.

3.2.2 Profile Ontology Learning

Content-centric SCSs offer well-organized contents which closely relate to user interest. In other words, we are able to collect high-quality training images for every node in the constructed profile ontology. For learning the profile ontology, we want to map the images curated by users onto this profile. For example, given an image of "cargo pants", we expect to *visually* reason like: $garment \rightarrow trousers \rightarrow pants \rightarrow cargopants$. Compared to the flat "bag-of-bundles" image organizations in Pinterest, this hierarchical reasoning gives richer semantic interpretations of user interest. In order to achieve this, for each node v in the profile ontology $\mathcal{O}$, we need to learn a classification model that predicts whether an image belongs to v. Let us start with looking for the training samples of v. Trivially, the images of the node itself will be the positive samples. Moreover, we consider positive samples of v's siblings as negative samples of v. This myopia way of training is shown to be effective in hierarchical visual task [82]. However, this training strategy suffers from the "error propagation" problem, *i.e.*, the models of v and its siblings are incapable of rejecting the classification errors propagated from higher-level unseen nodes.

In order to alleviate such propagated errors, we expect the model of every node in the ontology to be as accurate in prediction as possible. To achieve this, we propose to adopt the Multi-task Learning (MTL) framework [39] for jointly learning the models of a node and its siblings. It has been shown that MTL

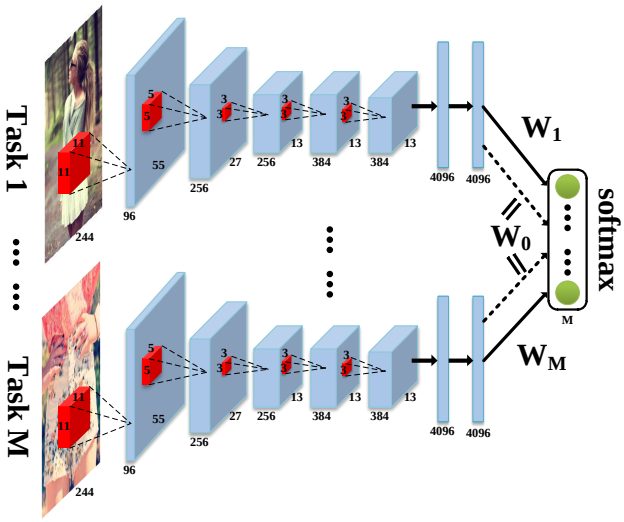


Figure 3.4: The illustration of the proposed mtCNN. There are M independent CNN pathways for the M tasks. These CNNs share a common parameter $\mathbf{w}_0$ when they eventually feed-forward into the softmax classification layer.

improves the prediction performance on multiple, different but related, learning problems through shared parameters or representations. In our case, the tasks of learning a node and its siblings are related. For example, "V-dresses" and "strapless dresses" under "dresses" share similar visual cues. Formally, without loss of generality, we only consider a set of sibling nodes $\{v_1, ..., v_M\}$, which share the same parent. Given training images $\{(\mathbf{x}_i, y_i)\}$, where $\mathbf{x}_i$ and y_i are the feature and label of the i-th image in any v_m, respectively. The objective of the MTL is

$$\begin{aligned} &\min_{\mathbf{w}_0,...,\mathbf{w}_M} F\left(\mathbf{w}_0, ..., \mathbf{w}_M\right) = \\ &-\sum_{m=1}^{M}\sum_{i\in\mathcal{I}_m} \log P\left(y_i = m|\mathbf{x}_i; \mathbf{w}_0, \mathbf{w}_m\right) + \lambda \sum_{m=0}^{M} R(\mathbf{w}_m) \end{aligned} \tag{3.2}$$

where $\mathbf{w}_0, ..., \mathbf{w}_M$ are the trainable parameters for M tasks, $\mathcal{I}_m$ is the set of training image indices of v_m, $P(y_i = m|\mathbf{x}_i)$ is a softmax function against other labels $y_i \neq m$ and λ is the trade-off parameter of the regularizer $R(\cdot)$, *e.g.*, ℓ_2-norm regularizer. Particularly, $\mathbf{w}_0$ is the shared parameters of the M tasks.

Recent advances in computer vision have shown that deep Convolutional Neural Network (CNN) can learn useful features that outperform the hand-crafted ones [69]. Therefore, we propose a novel multi-task CNN (mtCNN) deep architecture that jointly learns the features and parameters for the tasks.

As illustrated in Figure 3.4, we have M CNNs for each task to learn the feature $\mathbf{x}_i \leftarrow \phi(\mathbf{x}_i; \widetilde{\mathbf{w}}_m)$, where $\phi(\mathbf{x}_i; \widetilde{\mathbf{w}}_m)$ is the output of the m-th CNN (cf. Section 3.4.2 for the details of the mtCNN architecture) and $\widetilde{\mathbf{w}}_m$ is the trainable parameters. Denoting the the overall parameters of mtCNN as $\mathcal{W} = \{\mathbf{w}_0, \mathbf{w}_m, \widetilde{\mathbf{w}}_m\}_{m=1}^{M}$, then the stochastic gradient descent update rule of $\mathcal{W}$ in the k-th iteration of for solving mtCNN is

$$\begin{cases} \Delta_{k+1} = 0.9 \cdot \Delta_k - 1.5e^{-4} \cdot \eta \cdot \mathcal{W}_k - \eta \cdot \frac{\partial F}{\partial \mathcal{W}_k}, \\ \mathcal{W}_{k+1} = \Delta_{k+1} + \mathcal{W}_k, \end{cases} \tag{3.3}$$

where Δ is the momentum variable [100], η is the learning rate which is adaptive to the objective function value.

Now we can represent any image in the user-curated bundles as $\mathbf{p}$, where the i-th value p_i is the model output of item node v_i in ontology $\mathcal{O}$. By averaging the $\mathbf{p}$ of all the images, we can eventually obtain the user profiles in a hierarchical representation in which the value of each node shows the user's interest.

3.3 User Profiles Refinement by Social Curation

So far, the above user profiles are only based on visual models, which are insufficient to accurately predict user interest in terms of items in the ontology. In this section, we propose to refine the user profiles by exploiting multi-level social cues.

3.3.1 Formulation

We use graph links to model the various types of relations evident from rich social information of curated images. We observe that there are three levels of key relations in the content-centric social curation network. As we will detail soon, these levels of connections play an important role in regularizing the user interest depicted in images.

User-level. This level's connection origins from the fact that "Great minds think

alike". For example, if user A, B and C share images i and j simulataneously, then images i and j might be similar. Therefore, images are connected if two or more users have curated them. Formally, we have

$$S_{ij}^{u} = \begin{cases} n, \text{ if } n \text{ users share images } i \text{ and } j, \\ 0, \text{ no users share them.} \end{cases} \tag{3.4}$$

The strength of user-level link S_{ij}^{u} indicates how many users consider images i and j belong to the same interest.

Bundle-level. This level's connection is similar to the user-level connection since each bundle often represent one kind of a user's interest. Therefore, at this level, images are connected if two or more bundles include them,

$$S_{ij}^{b} = \begin{cases} n, \text{ if } n \text{ bundles include images } i \text{ and } j, \\ 0, \text{ no bundles share them.} \end{cases} \tag{3.5}$$

The strength of bundle-level link S_{ij}^{b} suggests how many bundles would be curated by users to pin images i and j to the same interest.

Content-level. This level includes two types of image content links: visual link and semantic link. The visual link is based on the visual similarities while the semantic link is based on the hierarchical semantic similarities between two images. Formally, we have

$$S_{ij}^{v} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{\rho^2}\right), \quad S_{ij}^{h} = \mathbf{p}_i^T \mathbf{H} \mathbf{p}_j = \sum_{k,l} \mathbf{p}_i^k H_{kl} \mathbf{p}_j^l, \tag{3.6}$$

where ρ is a predefined radius set to the standard variance of the feature norms, $\mathbf{x}$ and $\mathbf{p}$ are the visual and hierarchical semantic representations of images, respectively. The matrix $\mathbf{H}$ can be derived by measuring the closeness of item relations to the ontology. For instance, let $H_{kl} = \xi(\pi(k,l))$, where $\pi(k,l)$ is the lowest common ancestor of items k and l and $\xi(\cdot)$ is some real function that is non-decreasing going down the hierarchy, *i.e.*, the lower shared ancestor, the

more similar items k and l are.

Now, we can combine the above links in order to refine the existing hierarchical representations of images, which are the basis to establish user profiles. Denote the hierarchical representations of all the training images before refinement as $\mathbf{P}$ and those after refinement as $\mathbf{R}$, which is the goal we are pursuing. In particular, we assume that the original $\mathbf{P}$ is noisy and the desired $\mathbf{R}$ is a low-rank recovered noise-free matrix. Intuitively, each column vector of the low-rank matrix $\mathbf{R}$ denotes the hierarchical representation of an item. Intuitively, due to the relations of items in the ontology, the item "cargo pants" should imply that items "pants" and "trousers" are along the semantic path. This indicates, from the viewpoint of linear algebra, that "cargo pants" could be located in a subspace spanned by those items along the path, imposing a low-rank nature of the matrix $\mathbf{P}$.

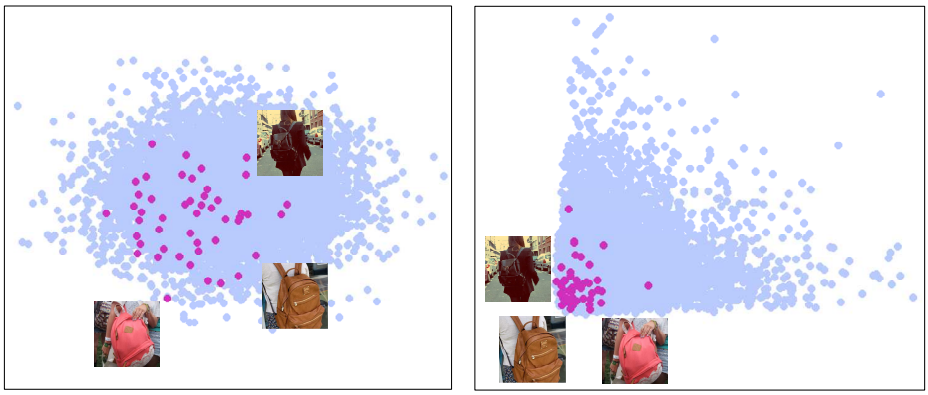
Therefore, by jointly considering the aforementioned multi-level social relations and the low-rank prior, the formulation of the proposed profile refinement objective is

$$\min_{\mathbf{R}} J(\mathbf{R}) = \|\mathbf{P}-\mathbf{R}\|_F^2 + \alpha\|\mathbf{R}\|_* + \beta \text{trace}\left\{\mathbf{R}^T\left(\mathbf{L}^u + \mathbf{L}^b + \mathbf{L}^v + \mathbf{L}^h\right)\mathbf{R}\right\} \quad (3.7)$$

where α and β are trade-off parameters, $\mathbf{L}^u$, $\mathbf{L}^b$, $\mathbf{L}^v$, and $\mathbf{L}^h$ are the graph Laplacians of the corresponding graphs in Eq. (4.3) to (4.4). For example, $\mathbf{L}^u = \mathbf{D}^u - \mathbf{S}^u$, where $\mathbf{D}^u$ is a diagonal matrix with the i-th entry as $\sum_j S^u_{ij}$. Such graph regularized terms impose the low-rank pursuit of $\mathbf{R}$ to be consistent with the multi-level social connections. The nuclear norm $\|\mathbf{R}\|_*$ is a convex surrogate for matrix rank [137], whose convexity allows an effective optimization for its solution. Next, we show how to solve the formulation in Eq 3.7.

3.3.2 Solution

When we investigate the formulation in Eq 3.7, we find that the profile refinement problem is a convex optimization problem. Therefore, there is a



(a) Original User Profiles (b) Refined User Profiles

Figure 3.5: Illustration of the effectiveness of the proposed profile refinement method. The gray blue points are images represented by user profiles, and the red ones correspond to "bag" images. All these points are visualized by using PCA mapped into a 2-D space. (a) Before refinement, images of the same interest are scattered. (b) After refinement, images of the same interest are clustered.

guarantee to obtain a global minimum. However, it does not have a closed-form solution. Fortunately, this problem can be solved by the Proximal Gradient method [126], which uses a sequence of quadratic approximations of the objective function $J(\mathbf{R})$ in order to derive the optimal solution.

We define $H(\mathbf{R}) = \|\mathbf{P} - \mathbf{R}\|_F^2 + \beta \text{trace}(\mathbf{R}^T \mathbf{L} \mathbf{R})$, where $\mathbf{L} = \mathbf{L}^u + \mathbf{L}^b + \mathbf{L}^v + \mathbf{L}^h$, and then the objective function can be re-written as $J(\mathbf{R}) = H(\mathbf{R}) + \alpha \|\mathbf{R}\|_*$. Suppose $\mathbf{R}_{k-1}$ is the solution at the $(k-1)$-th step, we can update to $\mathbf{R}_k$ by solving the following optimization problem which quadratically approximates $J(\mathbf{R})$ by the second-order Taylor expansion of $H(\mathbf{R})$ at $\mathbf{R}_{k-1}$:

$$\begin{aligned} \mathbf{R}_k &= \arg\min_{\mathbf{R}} H(\mathbf{R}_{k-1}) + < \nabla H(\mathbf{R}_{k-1}), \mathbf{R} - \mathbf{R}_{k-1} > \\ &\quad + \frac{\delta}{2} \|\mathbf{R} - \mathbf{R}_{k-1}\|_F^2 + \alpha \|\mathbf{R}\|_* \\ &= \arg\min_{\mathbf{R}} \frac{\delta}{2} \|\mathbf{R} - \mathbf{G}_k\|_F^2 + \alpha \|\mathbf{R}\|_*, \end{aligned} \tag{3.8}$$

where the values of $\mathbf{G}_k$ and δ are defined as

$$\begin{aligned} \mathbf{G}_k &= \mathbf{R}_{k-1} - \frac{2}{\delta} \left(\mathbf{R}_{k-1} - \mathbf{P} + \alpha \mathbf{L} \mathbf{R}_{k-1} \right), \\ \delta &= 2\sigma_{max} \left(\mathbf{I} + \alpha \mathbf{L} \right), \end{aligned} \tag{3.9}$$

where δ satisfies the Lipschitz condition and $\sigma_{max}(\cdot)$ denotes the largest singular values. Note that the solution of Eq. 4.5 is $\mathbf{R}_k = \mathbf{U} \text{diag} \lfloor \sigma - \frac{\alpha}{\delta} \rfloor_+ \mathbf{V}^T$, where $\mathbf{U} \text{diag}(\sigma - \frac{\alpha}{\delta}) \mathbf{V}^T$ is the singular value of $\mathbf{G}_k$ [137]. Also, note that even

with a large amount of images, the above optimization for profile refinement is tractable. To see this, for solving the singular values of $\mathbf{G}_k$, we can apply the trick to solve it by obtaining the singular values of $\mathbf{G}_k^T$, which can be much smaller. Meanwhile, for solving the largest singular values of $\mathbf{I} + \alpha\mathbf{L}$, we can adopt the efficient power method in matrix analysis.

After the above low-rank approximation, the average value of hierarchical representations of all the images after refinement $\mathbf{R}$ is regarded as the user profile. The effectiveness of this profile refinement algorithm is illustrated in Figure 3.5. It shows that the proposed methods can refine the user profiles with the same user interest (*i.e.*, "bag" in this example) close to each other so that they have consistent user profile representations. It gives an intuitive interpretation of the expected better recommendation performance of the proposed profile refinement since it is often much more accurate to calculate the user similarities, which are crucial in collaborative filtering approaches.

3.4 Experiments

In this section, we systematically evaluate the effectiveness of our proposed profile learning, profile refinement algorithm and image recommendation using the profiles.

3.4.1 Experimental Setup

DataSet

We crawled the Pinterest data based on HTTP requests as there is no open API of Pinterest. We followed traditional crawling protocol. We assume that popular pins represent the preferences of most active users on most popular topics. First, we started from 50 popular pins from seed data in fashion domain. Next, for each pinned image, we used a breadth-first search (BFS) strategy to crawl the boards which have pined or re-pined the image. The overall crawling process

took a month. As a result, the dataset consists of 1,239 users and 1,538,658 images.

In order to populate images into the constructed ontology in Section 3.2, we matched the user interest items (*e.g.*, bundle names, tags) with entries in our constructed ontology to annotate the images of the corresponding items. Besides, the time when the user pinned the images can be obtained, therefore, we divide the images based on the pinning time for image recommendation. In order to train the profile ontology model, we split the dataset into training/testing sets in half. Note that the testing set is used for testing ontology profile models, profile refinement algorithms and image recommendation. For profile refinement, the groundtruths are the as same as the profile models. For recommendation, the average number of images per user are 433. In order to simulate real-world recommender system, we added half noisy data (*i.e.*, around 200) that are not in fashion domain, to simulate the real recommendation system.

Compared Methods and Evaluation Metric

To evaluate the effectiveness of the proposed multi-task CNN (**mtCNN**), we compared it with state-of-art convolutional neural network [69]. For the model of each node in the ontology, we used the average precision (AP) as the evaluation metric.

To study the performance of our proposed profile refinement algorithm (**Ours**), two algorithms were employed as the baselines: a) **TRVSC** [76]: tag refinement algorithm based on visual and semantic consistency, b) **LR_ES_CC_TC** [148]: tag refinement algorithm low-rank, error sparsity, content consistency and tag correlation. For evaluation metric, we used the widely used F-score.

To evaluate the effectiveness of user profiling methods, we proposed to use image recommendation based on user profiling. As mentioned in [4], current recommender systems generally fall into the following two categories: a) content-based recommendations, where users are recommended items similar

to those they preferred in the past; and b) collaborative recommendations where users are recommended items that people with similar tastes and preferences liked in the past. Here, we extend the traditional recommendation methods using our proposed visual-based ontology profile. We used the content-based ontology profile vector to represent a user to calculate the user-item and user-user similarity. To evaluate the performance of visual-based profile in image recommendation, we compared it with state-of-art methods: a) **CB**: this is the traditional content-based recommendation method [98], where users are recommended items similar to those they preferred in the past; b) **CF_WNMF** [51]: This method makes the use of non-negative matrix factorization on user and item graphs for collaborating Filtering. c) **CF_LDA** [132]: This method combines traditional collaborating methods with probabilistic topic modeling to provide an interpretable latent structure for users and items. d) **CB_UP**: this method extends the traditional content-based methods through representing the users with our profile ontology, e) **CF_UP**: this method extends the traditional collaborating method by computing users' similarities using our proposed profile ontology. We used mean AP (mAP) of the recommendation results as the evaluation metric.

3.4.2 Implementation Details

The underlying deep architecture we adopted in Section 3.2 is the deep convolutional neural network architecture proposed by Krizhevsky *et. al* [69]. Its inputs were the raw RGB pixel intensity values of a 224×224 image. Those values were forwarded through 5 convolutional neural layers (with pooling and ReLU non-linearities activation function along the way) and 3 fully-connected layers to determine its final neuron activities, namely, a distribution over the sibling user interest items. As a result, neurons of the network in each layer were respectively 150,528-D, 290,400-D, 186,624-D, 47,996-D, 47,996-D, 43,264-D, 4,096-D, 4,096-D, and M-D, where M is the number of sibling items. We

used the ImageNet pre-trained model, namely DeCAF [35], as the pre-trained network to initialize CNN and the proposed mtCNN. For visual features of the images used in Eq. (4.4) and other content-based baseline methods, we also adopted the 6-th layer output of DeCAF, which is a 4096-D feature vector.

For the sparse coding in section 3.1, we empirically initialized λ as 0.1. For multi-task convolutional neural network, we empirically set λ as 0.0001. For the profile refinement algorithm, we set $\alpha \in \{0.0001, 0.01, 0.1..., 10\}$, $\beta \in \{0.00001, 0.01, 0.1, ..., 10\}$, various pairs of (α, β) values were tried and the one with the best performance was chosen.

For all the experiments, we used an NVIDIA 780X GPU with 2304 cores, 3GB memory, and i7-2600 CPU with 3.40 GHz and 16G memory.

3.4.3 Experimental Results

Evaluations of Profile Ontology Learning

Figure 3.6 illustrates the average precision values of different item classifiers in the hierarchy. From this result, we can see that the multi-task convolutional neural network (**mtCNN**) at most levels achieves a mean average precision about 0.50, which is superior to traditional neural networks. These results demonstrate the effectiveness of **mtCNN** that makes use of hierarchical visual tasks. However, it can be seen that our proposed method has comparatively low performance on some items such as "zipper front dresses" and "skinny pants". The reason for the low performance could be (a) the distinctive attribute of those items is too fine-grained, e.g. "zipper front", to differentiate those items correctly; and (b) these items tend to co-occur frequently with common items in an image. For example, the item "skinny pants" often co-occur with the item "high heels" in an image, then our method would recognize the "high heels" with "skinny pants" as "skinny pants". On the other hand, our method has quite good performance on other items such as "bamboo bag" and "goalkeeper glove". The reason for this is that those items have comparatively clean image

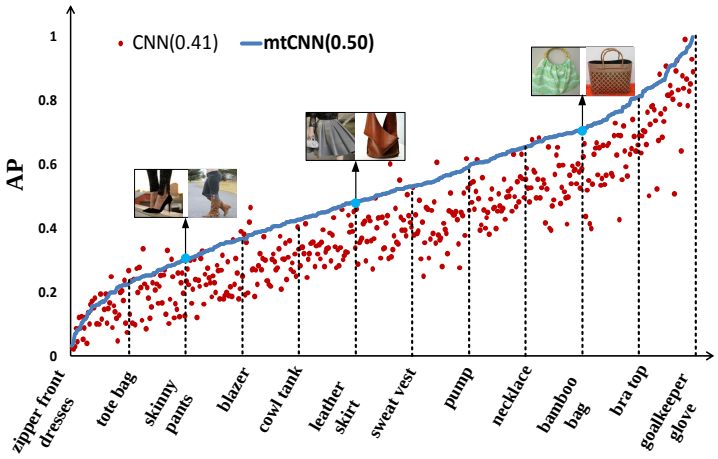


Figure 3.6: Performance of the 464 user profile models trained by CNN and the proposed mtCNN. mAPs are shown in brackets. Representative user interest and its two most confident images are shown.

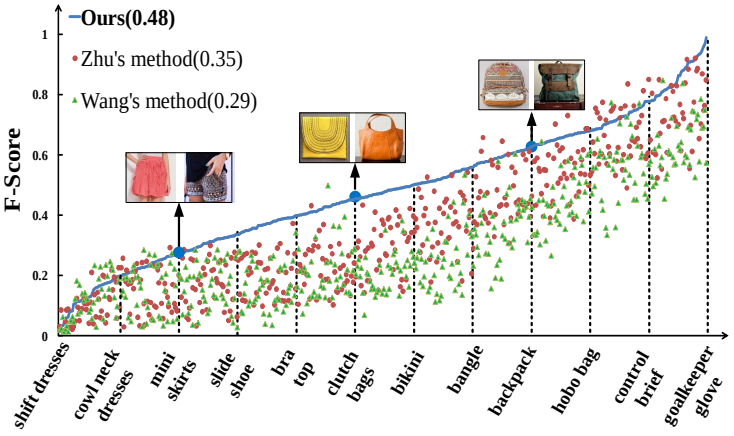


Figure 3.7: Performance of the three profile refinement methods. Mean F-Scores are shown in brackets. The profile contains 464 user interests. Representative user interest in the profile and its two most confident images are shown.

samples.

Evaluations Of Profile Refinement

Figure 3.8 shows some tag refinement results for some sample images produced by our approach. We can see that our approach can effectively correct and enrich the imprecise and incomplete image tags. For example, in Figure 3.8(c), our approach removes the irrelevant tags "dresses" and adds the more detailed tags "tote bag" and some other related tags such as "pink skirt" through social curation. Moreover, the enrichment capability of our proposed approaches can be seen in Figure 3.8(e) and (f), where there is only one irrelevant tag that

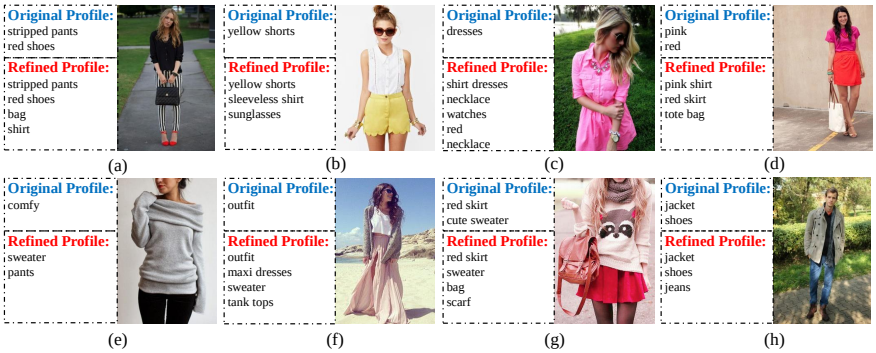


Figure 3.8: Illustrative profile refinement results by the proposed method.

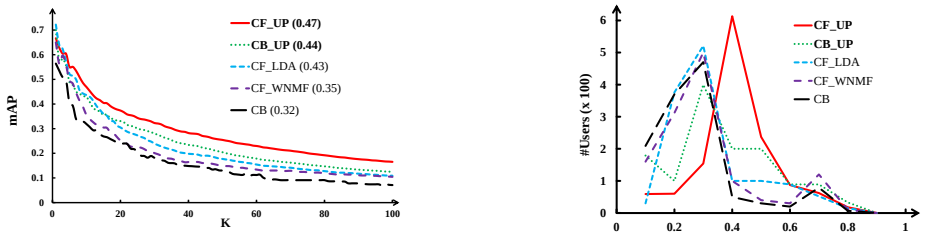
shows people's intentions such as "comfy" and "outfit" and after refining the incomplete tags by our approaches, the images are associated with reasonable tags. However, some case fail because of lack of sibling samples. For example, the "bow tie" and "ascot tie" under "tie" has fewer samples than other concepts on the ontology.

Figure 3.7 shows the detailed performance of image refinement for individual tags between our proposed approach and the baselines. From these results, we can see that the proposed method making use of social curation achieves an average F_1-score of 0.48, much higher than the other two methods. The superiority of our proposed profile refinement algorithms arises from two aspects:a) low rank; and b) multi-level social relations from social curations. Thanks to the content-centric network, multi-level content-content connections are encoded to refine the user profiles. The experiment results explicitly illustrate that social curation services provide more structured and accurate information to infer user's preferences.

From Figure 3.7, we can observe that some classes may have comparatively lower F_1-score. For example, the F-score of the item "mini skirts" is about 0.25. It may be due to the noisy content-level connection since "mini skirts" and "mini dress" are often pinned into the same boards. Moreover, some cases fail because of some image samples may not be that popular and therefore there exist sparse and noisy content-level connections. In contrast, we observe that items tend to have higher F-score performances that are popular in SCSs.

Evaluation of Image Recommendation

Figure 3.9(a) shows the performance comparison between the proposed image recommendation methods and the other four state-of-the-art recommendation methods. We can observe that our proposed recommendation methods based on the visual-based ontology profile achieve the best performance in terms of MAP at all the top K results as compared to the other methods. For example, our method improves the performance by 13.3%, 22.6%, 48.0%, 54.2% in terms of MAP at the top 20 results as compared to the **CB**, **CF_WNMF**, **CF_LDA** and **CB_UP** respectively. This verifies the effectiveness of our proposed ontology profile in recommendation systems. Figure 3.9(b) plots the user distribution under the five recommendation methods with mAP@10. We can see that our method can recommend the best images to most of the users. However, if the user's interest is too general, the framework does not work well since our recommender system will recommend all the images. Some illustrative examples are shown in Figure 3.10.



(a) (b)

Figure 3.9: (a) Performance (mAP@K) of the five recommendation methods. mAP@10 are shown in brackets. (b) The user distribution under the five recommendation methods with mAP@10

The superiority of our proposed ontology profile arises from the following points: a) This ontology profile models the user with a semantic hierarchy consisting of users' interest; such hierarchy profile provides a more comprehensive interpretation of images of interest of users; and b) through computing the users' similarities based on this hierarchy profile, implicit similar users can be obtained which alleviates the sharing sparsity problem in traditional

collaborative methods (*e.g.*, there may exist many images that are rarely shared and would not be recommended). Besides, since our ontology construction does not totally rely on Wikipedia, but also the user comments which cover the real interest even if it falls into the long tail. Once a general domain is selected, our ontology can adapt to the true distribution of user interest.

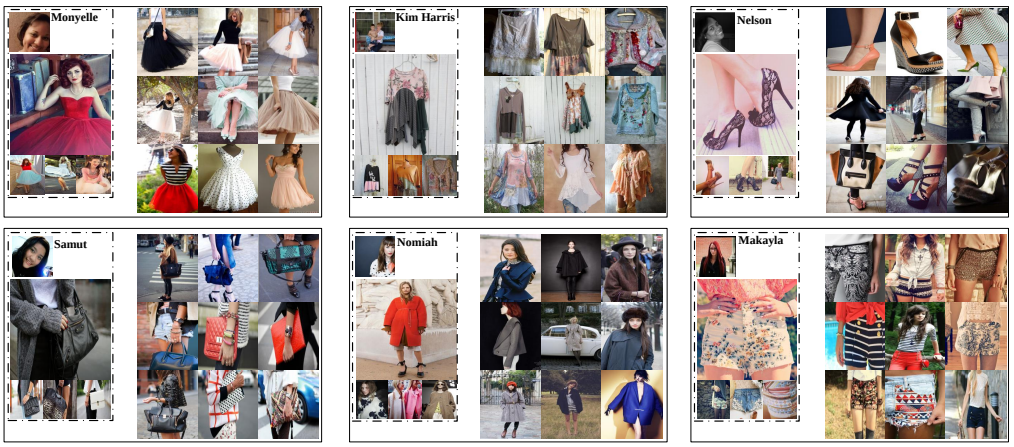


Figure 3.10: Illustrative recommendation results from the proposed collaborative filtering based on our user profiles (CF_UP). Top nine recommended images for six users are shown.

3.5 Summary

In this work, we targeted at content-based user profiling on the emerging social curation service (SCS). Compared to the conventional online social network service (OSNs), which focuses on user-user connections, SCS is based on the content-content connections curated by users. This is a unique characteristic of SCS, which inspires the idea of our profiling method. In particular, we investigated the fashion domain in the most popular SCS, namely Pinterest, on how social curation can help us tackle the existing difficulties in social media analysis. Specifically, we proposed to automatically construct a content-based user preference ontology and learn the ontological models to generate comprehensive user profiles. Then, we proposed to model the multi-level social relations offered by SCS to refine the user profiles in a low-rank recovery framework. Extensive experiments on 1,239 users and 1.5 million images

collected from Pinterest in fashion domain demonstrated the effectiveness of the proposed user profiling method, which outperforms the other state-of-the-art methods.

However, this work focused on profiling users in fashion domain. Next, we would like to introduce advanced methods to profile users in various domains based on the finding that the social connections between users and images play an import role in profiling users.

Chapter 4

User Profiling by Deep Learning of User-Media Interactions

In this chapter, we aim to propose a deep learning approach of breaking the user-media interactions into a deep hierarchy tree for learning user profiles. Furthermore, we exploit the observation of power-law user-media distributions to develop a synchronized optimization approach to improve the optimize speed of the proposed framework. Extensive experiments have demonstrated the effectiveness of the proposed approach.

4.1 Introduction

As mentioned in 1.1, the exponentially growing media contents will make it difficult for service providers to offer interesting products to specific consumers. This requires effective recommender systems to satisfy customers' needs.

However, the traditional recommender systems are not designed to function effectively in this new era of social curation marketing due to the following challenges: 1) The *extreme sparsity* of network structure (cf. Figure 1.4(a)). For instance, in Pinterest, an ordinary user often curates around 100 images which is only *one in a million* as compared to the whole Pinterest image collection. That is to say, it is hardly possible to infer the similarity between

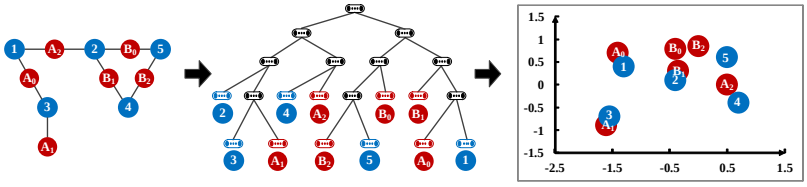


Figure 4.1: Our goal is to transform the users and images in a social curation network into a compact, low-dimensional feature space. Our approach takes user-image pairs in the social curation network (a) as input to the proposed deep architecture (b) which is built based on frequencies of user-image interactions. It then learns the user and image feature representations (c) as the output. Here is a simple illustration of our proposed method on a toy image-centric network: blue ones are users and red ones are images. We can see that the learned features can capture the pairwise user-image similarity.

users based on the shared images. Clearly, this will render collaborative filtering ineffective. 2) The *extreme diversity* of the multimedia contents (cf. Figure 1.4(b)). Different from products that can be easily categorized (such as those in Amazon), the categories of multimedia contents are usually hard to be identified automatically, causing difficulties for content-based recommender systems to infer accurate user interest from the curated contents, with the problem of over-specification [4].

In this work, we introduce a novel feature learning approach for recommendation that aims to tackle the above two extreme challenges in social curation. Different from conventional recommenders that indirectly rank images for users, we directly measure the similarity between users and images through a compact, low-dimensional vector space, spanned by "interest", which is the core motive of any social curation network. Our algorithm takes a social curation network with user-image links as input and produces latent representations of users and images as output. As illustrated by a toy network with 5 users and 6 images in Figure 4.1, we expect the vectors of linked users and images to be closer than other non-linked ones. The closer the pair of vectors, the higher the possibility that the user-image pair belongs to the same interest, and hence the rank of the image with respect to the user is higher.

Our model is a novel deep learning framework that breaks down a large and sparse network topology into a tree-structured deep hierarchy, where the

leafs are users and images (Figure 4.1). Each non-leaf feature encodes the information about the social interactions (*i.e.*, user-image, user-user, and image-image) and each resultant leaf embeds the "interest" of a user or an image into a vector. Note that our deep model is used as an "end-to-end" fashion, that is, we start from the most basic curation behavior "a user likes or dislikes an image" as the "low-level end", and the latent features forwardly propagate the curation belief into the resultant user-image features as the "high-level end". Different from shallow methods that attempts to learn user and image features directly [49; 68], our deep model can compactly [12] and efficiently learned representative features to reveal the weak correlations between images and users at the scene of the extreme sparse connections and extreme diverse images due to its deep structure.

In our proposed deep model, the input of user-image pairs could be over billions. Thus, how to efficiently optimize such a deep model becomes a big challenge. Fortunately, we observe that the user-image connections are long-tailed and very sparse, and hence there should be very limited shared parameters for different user-image pairs in the proposed deep tree structure. Therefore, we proposed a fast optimization algorithm that deploys an asynchronously parallel stochastic gradient descent method that can significantly reduce the time for the training of different user-image pairs.

We conduct extensive experiments on a representative subset of Pinterest, which is the most popular social curation network. In particular, the subset covers 468 popular interests on Pinterest with 1,456,540 images and 1,000,000 users who have interactions with these images. Through image recommendations, we demonstrate that the proposed deep model significantly outperforms the other state-of-the-art recommender systems. Our contributions are summarized as follows:

- We propose a deep learning framework for learning compact user and image features in a unified space from large, sparse and diverse social

curation networks. The learnt user and image features support effective recommendation by directly computing the similarity between the user vector and image vector.

- We develop a fast on-line algorithm to train the proposed deep learning framework. To our best knowledge, this is the first work on developing deep learning methods on content-centric networks.

The rest of the work is organized as follows. Section 4.2 describes the problem statement. Section 4.3 illustrates the proposed approach of deep learning structure. Experimental results and analysis are reported in Section 4.4, followed by summary in Section 4.5.

4.2 Problem Statement

4.2.1 Recommendation by Similarity

We consider the problem of recommending images denoted as $\mathcal{I}$ or users denoted as $\mathcal{U}$ to users in a social curation network denoted as $\mathcal{G} = \{\mathcal{U}, \mathcal{I}, \mathcal{E}\}$, where $\mathcal{E}$ is the set of edges that connect users and images. Although real-world social curation networks allow users to connect to other users[1], without loss of generality, we only assume that connections exist between users and images, *i.e.*, $\mathcal{E} \subseteq \mathcal{U} \times \mathcal{I}$. We are interested in the following user-image similarity measure:

$$s_{ui} = \mathbf{x}_u^T \mathbf{x}_i \tag{4.1}$$

where $s_{ui} \in \mathbb{R}$ is the rating score of image i being recommended to user u, $\mathbf{x}_u \in \mathbb{R}^d$ and $\mathbf{x}_i \in \mathbb{R}^d$ are the latent feature representations for users and images. In order to make a valid recommendation score by Eq. 4.1, we require $\mathbf{x}_u$ and $\mathbf{x}_i$ to encode interests. For example, if user u likes traveling and image i is about traveling, we expect the values of $\mathbf{x}_u$ and $\mathbf{x}_i$ to be consistently small.

[1]This rarely happens because most users only enjoy the curation function and ignore the social function.

In general, we seek a transformation $g : \mathcal{G} \mapsto \mathbb{R}^d$, where $\mathbb{R}^d$ is the unified space for users and images and thus facilitates direct user-image similarity measure in Eq. 4.1. Note that the transformation is generic since content-based filtering and collaborative filtering can be viewed in this form. For content-based filtering, it considers $\mathbf{x}_u$ as a content feature generated from the user's favored images. On the other hand, collaborative filtering treats $\mathbf{x}_i$ as the vector consisting of ratings $r_{u'i}$, where u' is a friend of u, and $\mathbf{x}_u$ is a vector of the similarities between the friends of u. As discussed in Section 4.1, the extreme connection sparsity and content diversity will make these traditional methods ineffective. For example, in content-based filtering, even if a user only likes a single interest "travel", it is difficult to generate $\mathbf{x}_u$ that is consistently similar to diverse images about traveling; in collaborative filtering, as the user-image connections are very sparse, it is impossible to infer accurate user similarities based on the shared images between users.

4.2.2 Modularity

Due to the sparsity of social networks, we wish to seek low-dimensional features for items (i.e.,, images) and users, through an objective that represents the interest communities of social networks. Modularity is a widely-used community partition measure that the larger the value, the better the partition of the network [28]. The underlying principle of using modularity is that the power-law distribution of connections between users and items is very significant in social curation network[2]. Consider the partitioning network $\mathcal{G}$ of n vertices (*e.g.*, $n = |\mathcal{U}| + |\mathcal{I}|$) and m edges into k non-overlapping interest communities. Let d_i represents the degree of vertex i. Modularity penalizes the situations when the number of within-group connections is smaller than the number of uniformly random connections, whose expected number is $d_i d_j / 2m$.

[2]The fraction of nodes in the network have k connections to other nodes is proportional to $k^{-\gamma}$.

Therefore, the modularity is formulated as:

$$J = \frac{1}{2m} \sum_{ij} \left(\mathbf{G}_{ij} - \frac{d_i d_j}{2m} \right) \delta\left(i, j\right), \tag{4.2}$$

where $G_{ij} = 1$ if i and j is connected and 0 otherwise, $\delta(i, j) = 1$ if i and j belong to the same membership and 0 otherwise. Note that $0 \leq d_i d_j / 2m \leq 1$, so the penalty comes in if $\left(\mathbf{G}_{ij} - \frac{d_i d_j}{2m} \right) < 0$. One aims to find a community partition over the network $\mathcal{G}$ when J is maximized. Note that we make no difference between users and items since our goal is to learn a unified space.

Although maximizing the modularity J over hard partition (*i.e.*, $\sigma(i, j) = 1$ or 0) is NP-hard [28], a relaxed approximation of the problem can be solved efficiently [124] when we relax the membership indicator $\sigma(i, j) = p\left(i|j\right) = exp\left(\mathbf{x}_i^T \mathbf{x}_j\right) / \sum_{i'} exp\left(\mathbf{x}_{i'}^T \mathbf{x}_j\right)$ as a valid probability: where $\mathbf{x}_i \in \mathbb{R}^d$ is a latent membership feature vector and the probability function is known as the softmax function. One can easily derive that this relaxed formulation is strongly related to the formulation of matrix factorization for recommendation [49; 68], which usually fails in sparse social network as we argued in Section 4.1.

4.3 Deep Learning Features for Social Networks

4.3.1 Architecture

In general, the latent interests encoded in the topology is difficult to be revealed by using these shallow methods when we directly solving Eq. 4.2. This is analogous to the situation in image classification, which suffers from the gap between noisy visual cues and the target labels. For this task, it is well-known that DCNN performs the best because they learn hierarchical features which are beneficial for the ultimate classification [12; 69]. Inherited from this core spirit of deep learning, we propose to solve Eq. 4.2 by a hierarchical deep model, which can learn useful intermediate features.

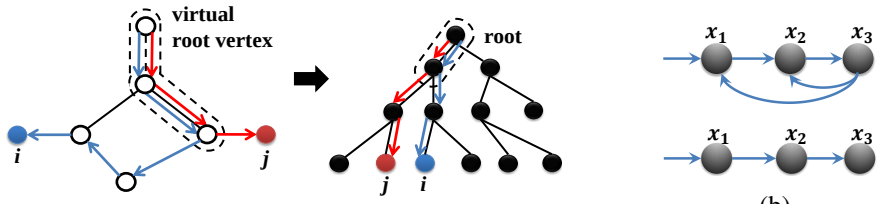
We start from introducing an approximation of $p(i|j)$ called "Hierarchical

Softmax", which is widely used in neural computation [93]. It approximates $p(i|j)$ by a series of binomial distributions along a tree-structured hierarchy. Specifically, we assign the vertices to the leafs of a binary tree (see Figure 4.1). For computation efficiency, the tree is a Huffman tree [88] according to the frequency of user-image interactions. Let $n_i(m)$ be the m-th node on the path from root to i, and let L_i be the length of this path. In particular, we have $n_i(1)$ as root and $n_i(L_i)$ as i. In addition, we denote $lc(n_i(m))$ as the left child of node $n_i(m)$ and let $I(n_i(m))$ be an indicator function such that it is 1 if $n_i(m+1) = lc(n_i(m))$, and -1 otherwise. Then, the hierarchical softmax version of $p(i|j)$ is defined as

$$p(i|j) = \prod_{m=1}^{L_i-1} \sigma\left(I[n_i(m)] \cdot \mathbf{x}_{n_i(m)}^T \mathbf{x}_j\right) \tag{4.3}$$

where $\sigma(x) = 1/(1+exp(-x))$ is the sigmoid function, which is widely-used to model the binary-valued binomial probability and $\mathbf{x}_{n_i(m)}$ is the representations of inner node $n_i(m)$. In terms of computation complexity, Eq. 4.3 can reduce the computation complexity of n sums (where $\mathcal{O}(n)$ can be millions in our case) with normalization in Eq. 4.1 to $\mathcal{O}(\log_2 n)$, which is significant.

Here, we show that the hierarchical softmax as formulated in Eq. 4.3 can be viewed as a deep architecture that represents the network topology. First, we can view the binary tree as a coding structure for each vertex in the network because each vertex i is assigned to a path from root to leaf. Then, the series of binary decisions from root to bottom mimic the route in the network from a common virtual root to vertex i. As shown in Figure 4.2(a), the route to the vertex i is by way of vertex j. The shared nodes along the path of j to i encode this routing information. So, we can view the nodes in the hierarchy encoding the topology of the entire network. Finally, we illustrate that Eq. 4.3 is in fact a forward propagation in the deep model. As illustrated in Figure 4.2(b), the difference between a traditional deep neural network and our network is that the proposed deep model is forwarded by using both the output features (*i.e.*, the leaf vectors) and the hidden units, while traditional neural network is forwarded by using



(a) (b)

Figure 4.2: Illustrations of the proposed deep architecture. (a) The node parameters of two paths in the deep hierarchy encode the topology information of a random walk from a virtual root to vertices. For example, the shared parameters correspond to the overlaps of the two routes (in dashed region). (b) Traditional deep architecture (bottom) feeds a fixed input into a forward network, while the proposed model (top) feeds both the output image and user features as input to every forwarding layer.

only the hidden units. Detailed information can be seen in Equation 4.5.

4.3.2 Formulation

We are interested in recommending image i to user u (or user u to image i). Intuitively, our learning objective seeks for feature representations $\mathbf{x}_u$ and $\mathbf{x}_i$ as

$$\begin{cases} \max\limits_{\mathbf{x}_u, \mathbf{x}_i} \ p(u|i) \text{ or } p(i|u), \text{ if } u \ likes \ i, \\ \min\limits_{\mathbf{x}_u, \mathbf{x}_i} \ p(u|i) \text{ or } p(i|u), \text{ if } u \ dislikes \ i. \end{cases} \tag{4.4}$$

Note that the above objective is consistent with the modularity maximization in Eq. 4.2. Moreover, we deploy a DCNN to transform images into the desired feature space: $\mathbf{x}_i = \text{CNN}(i)$, in order to generalize for new images. In this work, we adopt the AlexNet [69] where the softmax layer is removed but an additional fully-connected layer is added (i.e.,, from 4,096 to d neurons).

By incorporating the $p(u|i)$ formulated as in Eq. 4.3 into Eq. 4.2, the overall objective function becomes

$$\begin{aligned} \max_{\mathbf{x}_{n_u(m)}, \mathbf{x}_{n_i(m)}, \mathbf{x}_u, \text{CNN}(\cdot)} \\ J = \sum_{ui} A_{ui} \sum_{m=1}^{L_u - 1} \log \sigma \left(I[n_u(m)] \cdot \mathbf{x}_{n_u(m)}^T \text{CNN}(i) \right) \\ + \sum_{iu} A_{iu} \sum_{m=1}^{L_i - 1} \log \sigma \left(I[n_i(m)] \cdot \mathbf{x}_{n_i(m)}^T \mathbf{x}_u \right) \end{aligned} \tag{4.5}$$

where $A_{ui} = (G_{ui} - d_u d_i / 2m)$. Note that the above formulation allows us to encourage $p(u|i)$ to be larger if $A_{ui} \geq 0$ and smaller if $A_{ui} < 0$. Recall that

$0 \leq d_i d_j / 2m \leq 1$, so $A_{ui} \geq 0$ indicates user u likes image i and vice versa. Also, A_{ui} assigns a weight to encourage the connection G_{ui} if the expected connection $d_u d_i / 2m$ is small. For example, if user u is only linked to few images (*i.e.*, small d_u) and image i is only linked to few users (*i.e.*, small d_i), then an observation of u linked to i is informative. Therefore, the likelihood for $p(u|i)$ or $p(i|u)$ should be emphasized in optimization. For $A_{ui} < 0$, we only compute the pairs with the smallest 20 values for efficiency. Note that one can try more advanced negative sampling tricks [129], however, we found that there is no significant improvement.

4.3.3 Algorithm

For a typical social curation network, the number of user-image pair could be over billions. Therefore, it is impractical to optimize Eq. 4.5 even if we use the popular online stochastic gradient descent method for deep learning [12]. Here, we design a fast algorithm for tackling the large-scale networks. The main idea of our algorithm is that we deploy an asynchronously paralleled stochastic gradient descent method that can significantly reduce the time of scanning the user-image pairs.

The parallelization is made possible by the two observations from the structure of the topology parameters $\mathbf{x}_{n_u(m)}$ and $\mathbf{x}_{n_i(m)}$. First, as shown in Figure 3.5(a), the frequency distributions of users and images follow the power-law distribution. This observation is generally true in most social networks [95]. It means that we have a very long tail of infrequent pairs and thus the chance of two computing threads conflict when scanning the same pair is rare. Second, thanks to the binary tree structure of the parameters, the number of shared parameters between two leafs are limited. To see this, suppose that u and i correspond to sibling leafs, which is the worst case. The number of shared parameters is only $\log_2 n - 1$, where n is the total number of users and images. When $n = 10^7$, the fraction of affected parameters is only around 0.00002%,

which is negligible.

However, the parameters of CNN is shared by all the pairs. Therefore, jointly optimizing all the parameters in Eq. 4.5 will harm parallelization. To tackle this, we propose an alternative updating algorithm as shown in Algorithm 1. Specifically, we first fix the features of users $\mathcal{X}$ and CNN, and only update the topology parameters (*i.e.*, the inner node features) $\mathcal{T}$ as in Algorithm 2. Note that Steps 2-11 can be run asynchronously with multiple threads. In general, Algorithm 2 requires about 100 iterations for convergence. Next, as shown in Algorithm 3, we solve for $\mathcal{X}$ and CNN with fixed $\mathcal{T}$. It should be noted that $\mathcal{X}$ and CNN in Eq. 4.5 can be updated independently. In particular, they can be trained by asynchronous stochastic gradient descent on a distributed computing platform as described in [31]. We employ the momentum-based gradient descent as Steps 6-7 in Algorithm 2 and Steps 5-6 in Algorithm 3. This method has been shown to result in faster learning paces [100].

Algorithm 1: Deep Feature Learning for Images and Users

Input: Social curation network $\mathcal{G}$, feature dimension d
Output: User features $\mathbf{x}_u$ and image visual feature transformation CNN
1 **Initialization**: Build a binary tree for the users and images in $\mathcal{G}$; randomly set topology parameters $\mathbf{x}_{n_u(m)}$ or $\mathbf{x}_{n_i(m)} \in \mathcal{T}^{(0)}$, and user feature $\mathbf{x}_u \in \mathcal{X}^{(0)}$, initialize CNN with ImageNet pretrained model; randomly initialize the last layer of CNN, $t \leftarrow 0$
2 **repeat**
3 $\quad \mathcal{T}^{(t+1)} \leftarrow \text{UpdateTopology}\left(\mathcal{T}^{(t)}, \mathcal{X}^{(t)}, \mathbf{W}^{(t)}\right)$
4 $\quad \mathcal{X}^{(t+1)}, \text{CNN}^{(t+1)} \leftarrow \text{UpdateFeature}\left(\mathcal{T}^{(t+1)}\right)$
5 $\quad t \leftarrow t + 1$
6 **until** *converges*;

4.4 Experiments

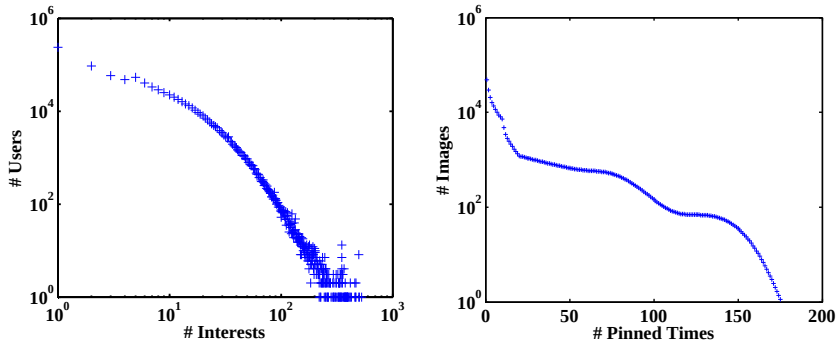
In this section, we conduct extensive recommendation experiments to evaluate the effectiveness of the learnt user and image features from the proposed model.

Algorithm 2: UpdateTopology $\left(\mathcal{T}^{(0)}, \mathcal{X}, \text{CNN}\right)$

Initialization: $t \leftarrow 0$, momentum $\Delta^{(0)} \leftarrow 0$, weight-decay factor α, learning rate η
repeat
  Online gradient descent:
  foreach *pair of* u *and* i **do**
    foreach $\mathbf{x} \in \mathcal{T}$ **do**
      $\Delta^{(t+1)} = 0.9\Delta^{(t)} - \alpha \cdot \eta \cdot \mathbf{x}^{(t)} - \eta \nabla_{\mathbf{x}} J(\mathcal{T}^{(t)})$,
      $\mathbf{x}^{(t+1)} = \Delta^{(t+1)} + \mathbf{x}^{(t)}$,
    end
  end
  $t \leftarrow t + 1$
until *converges*;
return $\mathcal{T}^{(t)}$

Algorithm 3: UpdateFeature $(\mathcal{T})$

Initialization: $t \leftarrow 0$, momentum $\Delta^{(0)} \leftarrow 0$, weight-decay factor α, learning rate η
repeat
  Stochastic Gradient descent:
  foreach *randomly selected mini-batch of user-image pairs* **do**
    $\Delta^{(t+1)} = 0.9\Delta^{(t)} - \alpha \cdot \eta \cdot \left(\mathcal{X}^{(t)}, \text{CNN}^{(t)}\right) - \eta \nabla_{(\mathcal{X}, \text{CNN})} J\left(\mathcal{X}^{(t)}, \text{CNN}^{(t)}\right)$,
    $\left(\mathcal{X}^{(t+1)}, \text{CNN}^{(t+1)}\right) = \Delta^{(t+1)} + \left(\mathcal{X}^{(t)}, \text{CNN}^{(t)}\right)$,
  end
  $t \leftarrow t + 1$
until *converges*;
return $\left(\mathcal{X}^{(t)}, \mathit{CNN}^{(t)}\right)$



(a) (b)

Figure 4.3: Pinterest dataset statistics. (a) This shows the number of users' interests; and (b) this shows the distribution of the times an image has been pinned.

4.4.1 Experimental Setup

Dataset

We used *Pinterest*, which is one of the largest social curation networks, as the source of the content-centric network for evaluating our proposed methods. To be noted that, the target of this Chapter is to exploit user-image connections for user profiling in general domain without domain knowledge. Hence, we collect a fresh representative data to validate our proposed method. In particular, given a user and his/her pinned images, we first found the category labels of these images and used these labels as the interests of this user. Specifically, the category labels come from Pinterest category site (e.g., `https://www.pinterest.com/categories/food_drink/`). We crawled the profiles of 1 million users together with their pinned images from Pinterest. The users were randomly sampled from the users communities found in the 468 categories we analyzed. For the pinned images, we removed images without category labels, resulting in 686,457 images. The remained user-image distribution is quite different from the dataset in Chapter 3. This is because that most images have comments which can help to annotate themselves in Chapter 3 while the dataset in this experiment only have rare labels annotated by Pinterest. We named this set of images I_u, those that actually pinned by users. In order to test the ability of recommending new images not pinned by users, we also crawled

additional 770,083 images which belong to the 468 interest categories but not pinned by any of the crawled users. The new image set is named as I_{new}. In the process, we also removed duplicated images which may impact the final evaluation results. These images were used to evaluate the performance of new image recommendation.

Figure 4.3 and Figure 3.5(a) show three distributions of our dataset: the distribution of the number of users' interests, the distribution of the times an image has been pinned, and the distribution of the number of users' pinned images. These distributions are power-law, where most users pin only a small number of images and have only a few interests; similarly, the images are only pinned by a very small number of users as compared to the total number of users. These distributions showed the sparsity and diversity of a typical social curation network. In order to demonstrate that our method can perform consistently well on different network topology, we randomly divided our dataset into 10 groups, each of which contains $100,000$ users and around $1,000,000$ images. The set of images includes those images pinned by the users in the group, with remaining randomly sampled from I_{new} set. The experiments were conducted on all the 10 groups. We reported averaged results with significance tests (applying t-test) and published the dataset [1].

Evaluation Metrics

We evaluated our method and other compared ones on image recommendation. We adopted the widely-used Normalized Discounted Cumulative Gain (NDCG) as the evaluation metric for both tasks. NDCG is defined as:

$$NDCG_k = \frac{1}{IDCG_k} \times \sum_{i=1}^{k} \frac{2^{r_i - 1}}{log_2(i+1)} \tag{4.6}$$

where $IDCG_k$ is the maximum $NDCG_k$ that corresponds to the optimal ranking list so that the perfect $NDCG_k$ is 1, and r_i is the degree of relevance

[1] https://sites.google.com/site/xueatalphabeta/academic-projects

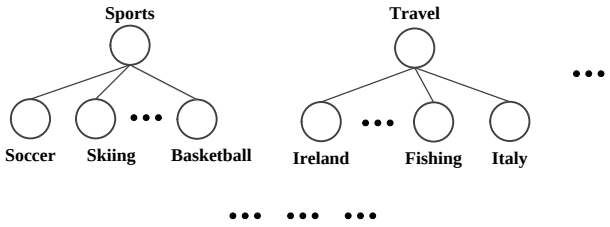


Figure 4.4: Interest categories in Pinterest are organized as a forest.

of the image in position i. We adopted a 3-scale $r \in \{0, 1, 2\}$ relevance score, representing *irrelevant*, *relevant*, and *highly relevant*, respectively. For image recommendation, we defined a recommended image to be: (a) highly relevant if the interest category of the image falls within the groundtruth interests of users; (b) relevant if the interest category of the image maps to sibling interests of users' groundtruth interests (Figure 4.4 illustrates a part of the interest category forest collected from Pinterest); and (c) irrelevant if none of the above.

In addition to NDCG which measures the relevance of the recommended images, users may also prefer the recommended images to be more diverse, *i.e.*, if a user has many interests, results that cover more interests are preferred. Therefore, we used entropy $H_k = -\sum_{i=1}^{R} p_i \ln p_i$ to measure the diversity of the recommendation results, where S_k is the set of successfully recommended (highly relevant and relevant) images up to position k, R is the total number of types of interests in S_k, and p_i is the proportion of images belonging to the ith type of interest in S_k. Here, a larger H_k represents more diverse results.

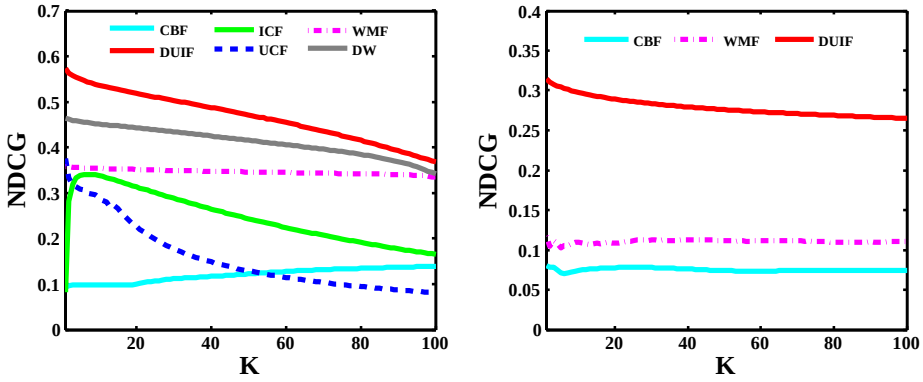
Comparing Methods

We compared the performance of our proposed Deep User-Image Feature (**DUIF**) with the following five baseline methods: a) Content-based filtering (**CBF**) [98; 119]: It generates a user feature vector by averaging all the image features (we used the state-of-the-art 4,096-d DeCAF [36] feature) pinned by the user and then recommend images based on the similarity between the image features and the user features. b) User-based collaborative filtering (**UCF**) [145]: It analyzes the user-image matrix to compute the similarities

between users and then recommends images to people with similar tastes and preference. c) Item-based collaborative filtering (**ICF**) [32]: This technique first analyzes the user-image matrix to identify relationships between different images, and uses these relationships to indirectly compute recommendations for users. d) Weighted Matrix Factorization (**WMF**) [129]: It decomposes the user-image matrix into latent user and image features by the weighted matrix factorization [59] and uses CNN to regress images to the image vectors. e) Deep Walk (**DW**) [99]: It learns the user and image latent representations of vertices in a social network by applying a language model. Then, images are recommended by the similarity between the user features and the image features. We empirically tested different configurations of baseline methods and employed the best ones as baselines.

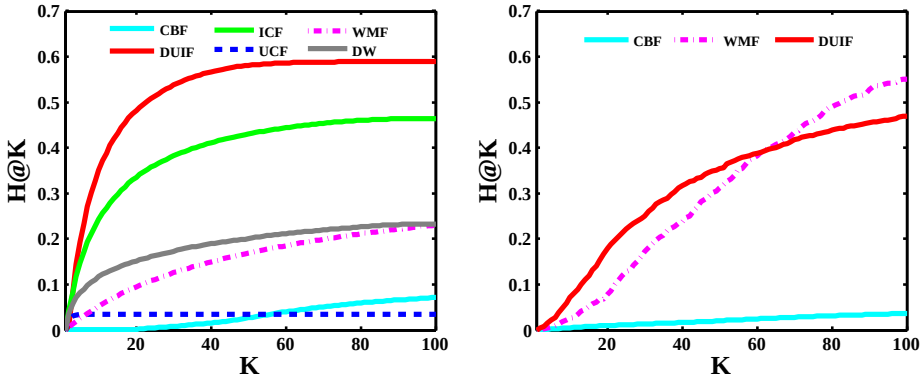
4.4.2 Implementation Details

For deep CNN, we depoyed Caffe framework [62] for CNN implementation on a NVIDIA Titan Z GPU. In particular, we used the well-known AlexNet architecture [69], which consists of 5 convolutional layers with max-pooling and 2 fully connected layers before the loss layer. Our CNN added an additional fully connected layer for the resultant d-dimensional feature space. We used the author provided ImageNet pretrained model (in Caffe format) as initializations. The initial learning rate was set to $1e^{-4}$ with dynamic momentum. The size of the batch was 128 and it took 20 epochs to converge using Algorithm 3. Each epoch took about 40 mins. For Algorithm 2, we randomly initialized all the parameters, and the starting learning rate was set to $1e^{-5}$ with dynamic momentum. We used 8 computing threads on a 8-core machine. It took around 100 epochs to converge with each epoch taking about 10 mins. For the above algorithms, we used ℓ_2-norm weight decay with $5e^{-5}$ coefficient. For Algorithm 1, we found that 2 iterations were sufficient for a good solution. The choice of feature dimension is crucial. We tuned the values within $\{100, 200, ..., 1,000\}$



(a) (b)

Figure 4.5: Performances ($NDCG_k$) of various methods on recommending new images to users based on (a) existing pinned set (I_u) and (b) new image set I_{new}.



(a) (b)

Figure 4.6: Performances of diversity (H_k) of various methods on recommending new images to users based on (a) existing pinned set (I_u) and (b) new image set (I_{new}).

and found that 300 was the best choice.

4.4.3 Experimental Results

For our evaluation, we want to test the effectiveness of the recommendation methods to recommend new images based on those pinned by existing user community I_u and those unseen images I_{new} not pinned by existing user community. We note that among the five baseline methods, CBF is based on the contents of the images, UCF and ICF are traditional collaborative filtering methods, while WMF and DW are based on latent factors. We note that UCF, ICF and DW cannot be used to recommend new images, which are unseen in existing networks. Hence for testing recommending new unpinned images from

Table 4.1: Detailed recommendation performance ($NDCG_k$) on recommending new images to users based on existing pinned set (I_u) and new image set (I_{new}) with significance test. Results labeled with ‡ are highly significant (p<0.01), and † indicates significant (p<0.05), against the best comparing method.

Existing Image Recommendation					
	$NDCG_5$	$NDCG_{10}$	$NDCG_{20}$	$NDCG_{50}$	$NDCG_{100}$
CBF	0.098	0.099	0.100	0.122	0.139
UCF	0.308	0.290	0.226	0.129	0.081
ICF	0.338	0.338	0.314	0.244	0.165
WMF	0.356	0.354	0.352	0.346	0.334
DW	0.457	0.451	0.443	0.416	0.342
DUIF	**0.550**‡	**0.537**‡	**0.519**‡	**0.472**‡	**0.368**†
New Image Recommendation					
	$NDCG_5$	$NDCG_{10}$	$NDCG_{20}$	$NDCG_{50}$	$NDCG_{100}$
CBF	0.079	0.080	0.081	0.080	0.081
WMF	0.103	0.110	0.108	0.111	0.110
DUIF	**0.304**‡	**0.298**‡	**0.289**‡	**0.276**‡	**0.265**‡

Table 4.2: Detailed recommendation performance (H_k) on recommending new images to users based on existing pinned set (I_u) and new image set (I_{new}) with significance test. Results labeled with ‡ are highly significant (p<0.01), and † indicates significant (p<0.05), against the best comparing method.

Existing Image Recommendation					
	H_5	H_{10}	H_{20}	H_{50}	H_{100}
CBF	0.000	0.000	0.002	0.027	0.071
UCF	0.034	0.035	0.035	0.035	0.035
ICF	0.147	0.243	0.335	0.430	0.465
WMF	0.025	0.052	0.095	0.169	0.230
DW	0.082	0.117	0.152	0.201	0.233
DUIF	**0.194**‡	**0.350**‡	**0.481**‡	**0.581**‡	**0.589**†
New Image Recommendation					
	H_5	H_{10}	H_{20}	H_{50}	H_{100}
CBF	0.002	0.005	0.010	0.020	0.037
WMF	0.005	0.025	0.078	0.312	0.551
DUIF	**0.022**‡	**0.071**‡	**0.180**‡	**0.354**‡	**0.470**

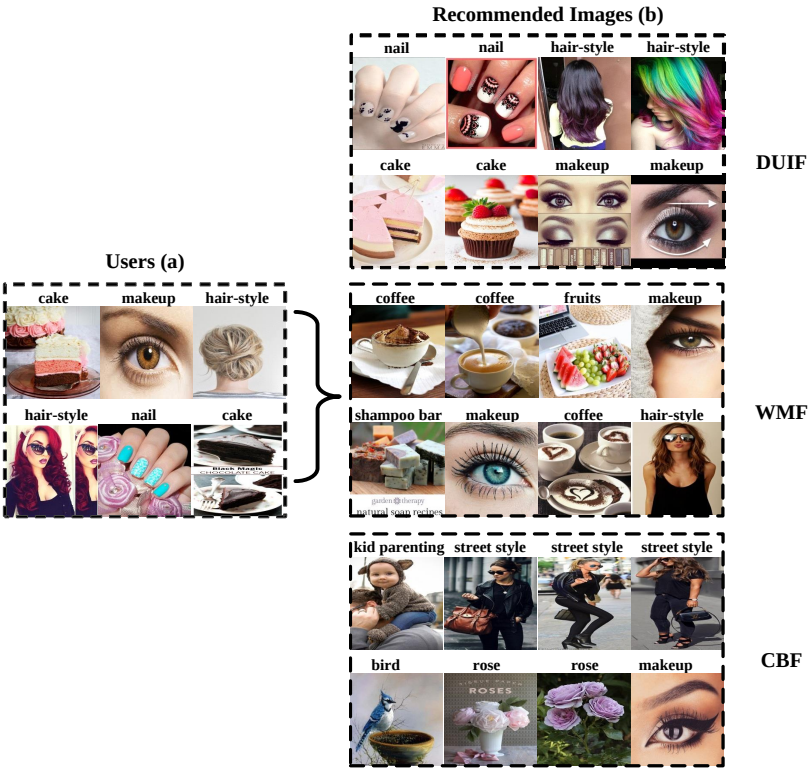


Figure 4.7: Illustrative examples of recommending new images to users using different methods (b) based on users' pinning profiles (a).

set I_{new}, we only compare our proposed method with CBF and WMF.

Figure 4.5 and 4.6 compare the performance of recommendation methods to recommend relevant images to users based on existing pinned set I_u and new image set I_{new}. Figure 4.5 presents the performance in terms of relevance based on $NDCG@K$; while Figure 4.6 presents the performance in terms of diversity based on $H@K$. In addition, Table 4.1 and Table 4.2 separately lists the respective results with significant test on image recommendation at the top 5, 10, 20, 50 and 100 positions. Some illustrative examples are shown in Figure 4.7.

As can be seen from the results, the proposed DUIF significantly outperforms the other methods for image recommendation. The comparatively good performance of DUIF mainly comes from the following aspects. As previously introduced, the multimedia contents are very diverse, even for the same interest topic, hence methods (*e.g.*, CBF) that only consider image contents would have poor performance. Moreover, each user often has many different interests. Such diverse images and varying users would result in a more sparse and complex

user-item matrix, which renders those matrix decomposition based methods such as UCF and WMF ineffective in revealing the underlying user interests. Further, we observe that the latent factor based models such as WMF often outperforms the traditional collaborative filtering methods such as UCF and ICF. The findings verified that methods that attempt to discover compact latent vectors for users and images tend to perform better than those that directly apply the user-image matrix. Finally, although DW which is similar to DUIF, it does not consider the contents of images and the intrinsic property of social curation network, namely modularity. Hence it performs worse than DUIF on the recommendation task. Overall, DUIF differs from the baseline methods in that it jointly considers image content analysis and social curation network topology. Experimental results have shown that it can effectively map images and users into a unified space for effective image recommendation.

4.5 Summary

We proposed a novel deep learning framework for learning the representations for topological user nodes and visual images in large, very sparse and diverse social curation network and applied the resulting model to recommender system. Experimental results on a representative subset of Pinterest with about 1.4 million images and 1 million users have demonstrated that the proposed approach can significantly outperform other methods. Exploiting social media data to generate features could be a promising research direction in computer vision community. Furthermore, in the proposed deep architecture, the rich textual information which can provide important semantic meanings for images has not been exploited. In the future, we can further investigate the case of mining the multi-modal contents of texts and images for the task of user profiling.

Chapter 5

User Profiling by Integrating Knowledge, User-Media Interactions and Multi-modal Contents

Until now, we have conducted research on exploring domain knowledge and user-media interactions to construct user profiles, respectively. Further, this chapter aims at integrating knowledge, user-media interactions and multi-modal contents together to infer user profiles for personalized services. In particular, the proposed approach incorporates both user-media interactions and media-text associations to learn compact representations guided by the domain knowledge for users, rich media, textual information simultaneously.

5.1 Introduction

To now, we have addressed the problem of user profiling using multimedia contents generated by users and content-centric network. However, the task of user profiling still faces several challenges as below:

Figure 5.1: We wish to learn out a latent visual-based and semantic-based user profile. For instance, given the fashion products shared by a specific user, our proposed model extracts the user's semantic interest such as "Chanel", "Fosiil" and "dress", and a visual latent-based vector that shows the user's preferences. Based on such learnt profile, we can conduct image recommendation effectively and efficiently.

- **Representation of User Profiles** Chapter 3 and Chapter 4 concentrate on construct semantic-based user profiles and latent-based user profiles separately without a composite of them. In this work, we attempt to combine the user generated multi-modal contents to infer a semantic-based and latent-based user profile.
- **Analysis of Heterogenous Data** Online social networks (OSNs) are heterogenous in nature where consumers share multi-modal contents with each modality expresses partial view of users' interest [8]. Most existing studies [69; 123; 24] learn features of contents by harvesting uni-modal and multi-modal information without incorporating users, little efforts have been made on jointly embedding of users and multi-modal contents. Luckily, the remarkable *collective intelligence* [111] can help us tackle this problem. For example, if many users share the same image "Steve Jobs", they may have the same interest of admiring "Steve Jobs". Therefore, how to effectively learn the jointly embedding of users and multi-modal contents using *collective intelligence* remains a challenge.
- **Utilization of Human Prior Knowledge** As mentioned by 1.1.1, human prior knowledge plays an important role in the user profiling process and has been proved its importance in Chapter 3. Hence, this work go deeper

to explore what kind of human prior knowledge can be exploited and how to incorporate the knowledge into our model.

We tackle the above issues by developing a novel model, named, the Embedded Learning of Users, Contents and Knowledge (**EmLUCK**), to profile users in a latent-based and semantic-based way. An example is shown in Figure 5.1. Based on users' shared contents, the proposed model will learn a semantic-based profile including concepts like "Chanel", "Fosil" and "watch" and a visual latent-based user profile showing the user's visual latent preferences that can be used to directly and efficiently support product recommendation.

Note that we take apparel domain as an example. One of the most basic principles of fashion design and styling is color [18], especially the theory of *color harmony*. Putting together a set of harmonious colors can produce a pleasing affective response to users [22]. *Color harmony* is nothing more than time-tested recipes, as they were, for colors that work well together, they tend to be related or contrasting [41]. It has been widely employed recently in fashion industry. For instance, stylists from Vogue have advocated to apply *color harmony* in their portraits [1] while Chanel has employed the *color harmony* into their eyeshadow products [2]. Indeed, understanding *color harmony* would lead to a well-thought mix of outfit. For instance, the blue T-shirt goes well with an orange tie as shown in Figure 5.4(a), as blue and orange colors are in the complementary scheme. Such time-tested principle of *color harmony* which is different from traditional color histogram representation, is hardly understandable by machines. Additionally, we need a semantic-based user profile to support reasoning [2]; such profile also cannot be figured out by machines. Luckily, we can incorporate the clothing ontology that structures these semantic information [133] to achieve the goal.

Driven by collective intelligence, we propose a matrix factorization approach, **EmLUCK**, that explores the heterogenous networks of contents and users,

[1] http://www.vogue.it/en/talents/talents-shooting/2012/11/colors-in-harmony#ad-image235356
[2] http://les4ombres.chanel.com/en_SG/harmonies

guided by the human prior knowledge. **EmLUCK** is able to map users and the multimedia contents they shared into a common low-dimensional space. Consequently, the recommendation of images to users can be conducted by directly measuring the similarity between users and images; and friend recommendation can also be done in a similar way. Moreover, by measuring the similarity between users and texts, a semantic-based user profile can also be constructed.

We conduct extensive experiments on the Amazon apparel dataset and a representative subset of Pinterest, which is the most popular social curation network. Through image-based fashion product recommendation, we demonstrate that the proposed method significantly outperforms existing state-of-the-art recommenders. The contributions of this study are:

- We develop a novel method that integrates multi-modal contents shared by users to build a visual-based and semantic-based user profile.
- We explore collective intelligence existed in the heterogenous networks of users and contents, and integrate color harmony from visual Art and clothing ontology into the proposed model to improve the recommendation performance. To our best knowledge, this is the first work to employ color theory from visual Art for user profiling in apparel domain.

The remainder of the chapter is structured as follows. Section 5.2 details the issue we attempt to solve while Section 5.3 introduces our model. The experiment is detailed in Section 5.4.

5.2 Problem Statement

Let $\mathcal{U} = \{u_1, u_2, ..., u_{N_u}\}$ be a set of users, $\mathcal{V} = \{v_1, v_2, ..., v_{N_v}\}$ be the set of their shared images and $\mathcal{T} = \{t_1, t_2, ..., t_{N_t}\}$ be the set of keywords extracted from comments or source links where N_u, N_v and N_t are the numbers of users, images and keywords, respectively. A user u_i can share/buy an fashion-related

product image v_j from the source link and attach a comment. We use the matrix $\mathbf{M} \in \mathbb{R}^{N_v \times N_u}$ to denote the image-user association matrix where $\mathbf{M}(i,j) = 1$ if image v_i is shared by user u_j, otherwise zero. $\mathbf{Q} \in \mathbb{R}^{N_v \times N_t}$ is the image-keyword matrix where $\mathbf{Q}(j,k) = 1$ if image v_i is associated with keyword t_k and $\mathbf{Q}(j,k) = 0$ otherwise. Moreover, $\mathbf{X} \in \mathbb{R}^{N_v \times D}$ is the content features of $\mathcal{V}$. Additionally, note that this work takes apparel domain as an example. We denote human prior knowledge as $\mathbf{\Omega}$. Since we propose to map users, images and texts in a common-low dimensional space, we denote the learnt latent vectors of users, images and keywords as $\mathbf{R}, \mathbf{I}$ and $\mathbf{H}$. The involved notations of this work are summarized in Table 5.1.

User Profile: A user profile is formally defined as two vectors in this work. One weighted vector $\mathbf{f}_1$ shows the semantic interest while another vector $\mathbf{f}_2$ shows the latent visual-based preferences of the user.

With the aforementioned defined notations and definitions, the problem of user profiling can be stated below:

Given N_u users, image-user association matrix $\mathbf{M}$ and image-keyword matrix $\mathbf{Q}$ as well as domain knowledge $\mathbf{\Omega}$, we aim to develop a method f, which can generate a semantic interest vector $\mathbf{f}_1$ and a latent factor on visual preferences $\mathbf{f}_2$ for user u_i for recommendation by learning a latent low-dimensional common space $\mathbf{R}, \mathbf{I}$ and $\mathbf{H}$ of users, images and keywords separately.

$$f : \{\mathbf{f}; \mathbf{M}, \mathbf{Q}, \mathbf{\Omega}\} \xrightarrow{\mathbf{R},\mathbf{H},\mathbf{I}} \{\mathbf{f}_1, \mathbf{f}_2\}_{u_i} \tag{5.1}$$

5.3 Embedding of Heterogenous Networks

As shown in Figure 5.2, we attempt to learn the embedding of users, images and knowledge respectively by mining the heterogenous associations of user-content and human prior knowledge *i.e.*, color harmony and clothing ontology. Such learnt embedding can be deployed in many applications such as

Table 5.1: Definition of notations

Notations	Descriptions
$\mathcal{U}, \mathcal{V}, \mathcal{T}$	set of users, images and keywords, respectively
N_u, N_v, N_t	number of users, images and keywords, respectively
$\mathbf{M} \in \mathbb{R}^{N_v \times N_u}$ $\mathbf{Q} \in \mathbb{R}^{N_v \times N_t}$	image-user matrix and image-keyword matrix, respectively
D	dimension of image content features
l	dimension of learnt latent feature space for users, images and keywords
$\mathbf{X} \in \mathbb{R}^{N_v \times D}$	content features of images
$\mathbf{R} \in \mathbb{R}^{N_u \times L}$ $\mathbf{I} \in \mathbb{R}^{N_v \times L}$ $\mathbf{H} \in \mathbb{R}^{N_t \times L}$	latent factor matrix of users, images and keywords, respectively
$\mathbf{W} \in \mathbb{R}^{D \times L}$	matrix correlated image content features with latent factors
Γ	a set of similar or matchable pairwise images
Λ	clothing ontology
O	a set of harmony color schemes
S	a set represents a specified color scheme
α, β, γ	weights of different components
λ	regularizer penalty coefficients

recommendation, community detection and topic detection. In the following subsections, we will illustrate the key steps in building the proposed model in terms of heterogenous networks of user-contents and prior knowledge, followed by detailed optimization.

5.3.1 Collective Representation Learning

One of the most popular approaches to model such associations between different modalities is matrix factorization by characterizing both users and items into vectors of latent factors. The goal of matrix factorization is to map images and users into a latent space of dimension l in which image-user interactions are modeled as inner products in the latent space as:

$$\mathbf{M} \approx \mathbf{I}\mathbf{R}^T \tag{5.2}$$

where $\mathbf{R} \in \mathbb{R}^{N_u \times l}$ and $\mathbf{I} \in \mathbb{R}^{N_v \times l}$ are latent representations of users $\mathcal{U}$ and images $\mathcal{I}$ separately. Each row $\mathbf{I}_i$ represents a community of users that are interested in such latent space of image v_i; while each row $\mathbf{R}_j$ shows a set of images that expresses the preference of user u_j in the latent space.

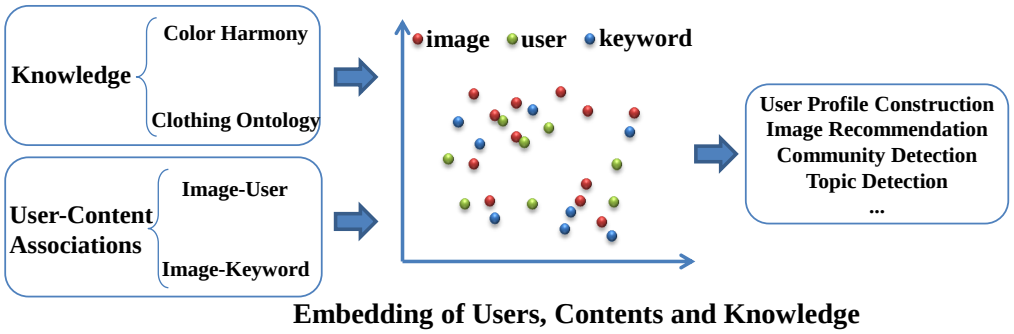


Figure 5.2: A framework for collective representation learning with prior knowledge.

For each image, in addition to representing an image as a hidden community of users in Eq. 5.2, we also think of embedding the image in terms of the hidden topics derived from the image's text reviews, descriptions or source link. Then we have:

$$\mathbf{Q} \approx \mathbf{I}\mathbf{H}^T \tag{5.3}$$

where $\mathbf{H} \in \mathbb{R}^{N_t \times l}$ is the representation of keywords. Each row $\mathbf{I}_i$ serves as a set of keywords denoting the contents of image v_i in the latent space; while each row $\mathbf{H}_j$ expresses the semantic meaning of a keyword t_j using a set of images in the latent space.

The key assumption in our formulation is that we have a common decomposition matrix $\mathbf{I}$ for both Eq. 5.2 and Eq. 5.3. The $\mathbf{I}$ matrix acts as a bridge to connect the two disparate components. This assumption comes from the notion of collective intelligence, generally referred as *collective factorization* [65; 111], and usually encloses a common variable over different modalities. Such notion of collective intelligence has been employed in many applications such as link prediction [111]. It can be explained in this way that a particular set of users will be dedicated to a particular topic via an image. Therefore, we can decompose an image in terms of its topics or its communities in the same manner. For example, an image about "Steve Jobs" can be considered as 50% for famous people "Steve", 30% for "Apple" related company and 20% for spreading across other relevant topics. Another facet is that different communities of users may have different aspects of interest in "Steve". Accordingly, the same image can

be equivalently broken down as 40% for community interested in the brilliant "Steve", 40% for community interested in the product of "Apple", and 20% disseminated to other communities. Eq. 5.2 and Eq. 5.3 figure out the two distinctive components to our objective function.

Another common issue in social recommendation is the *cold-start* problem [147], in which the recommendation performance suffers for items with few or no prior ratings/views. Approaches that do not consider content features of items will fail. Inspired by the work of [129] which embeds deep content feature into music recommendation and [147] which brings content items into their latent model for user profiling, we tie in the content feature of images $\mathbf{X}$ to address this issue. Hence, the objective function can be concluded by the following equation:

$$\begin{aligned}\min_{\mathbf{W},\mathbf{H},\mathbf{R}} J =& \frac{1}{2}\| \mathbf{M} - \mathbf{X}\mathbf{W}\mathbf{R}^T \|_F^2 + \frac{\alpha}{2}\| \mathbf{Q} - \mathbf{X}\mathbf{W}\mathbf{H}^T \|_F^2 \\ &+ \frac{\lambda}{2}(\| \mathbf{W} \|_F^2 + \| \mathbf{R} \|_F^2 + \| \mathbf{H} \|_F^2)\end{aligned} \tag{5.4}$$

where $\| \cdot \|_F$ is the Frobenius norm of a matrix and each row $\mathbf{X}_i$ is the content feature of an image v_i; while $\mathbf{W} \in \mathbb{R}^{D \times L}$ is a matching matrix that correlates the content features of images with users $\mathcal{U}$ and keywords $\mathcal{T}$. Obviously, the latent features of images $\mathbf{I}$ in Eq. 5.2 and Eq. 5.3 is replaced by a multiplication of the image content features $\mathbf{X}$ and the transformation matrix $\mathbf{W}$. α is introduced to leverage the contribution of image-keyword matrix. In practice, it is common to put regularization penalties on $\mathbf{W}, \mathbf{R}$ and $\mathbf{H}$ to avoid over-fitting.

5.3.2 Utilization of Prior Knowledge

There are evidences that human knowledge can be used to improve the performance of many applications [133]. Since the final consumer is human, human interpretations of recommendation are important and shall be utilized in the proposed model. Taking the apparel domain as an example, we exploit the color harmony and clothing ontology to improve the user profiling performance.

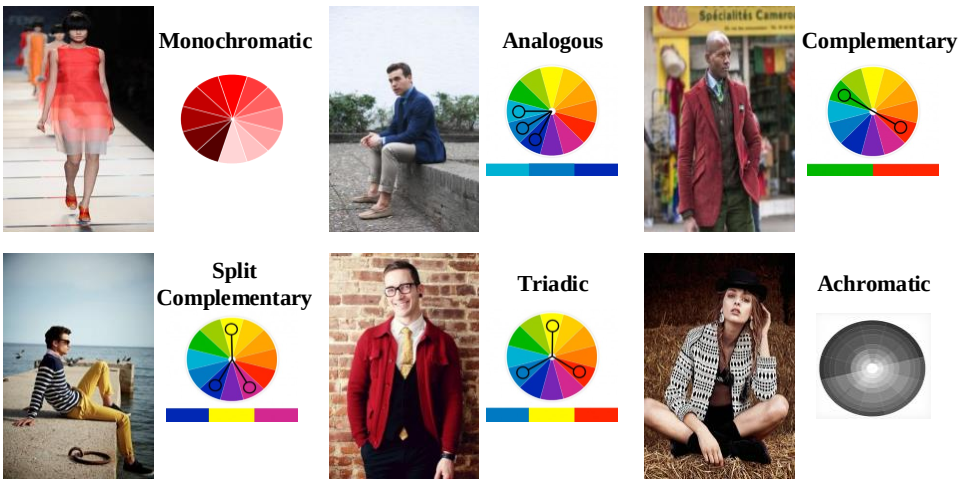


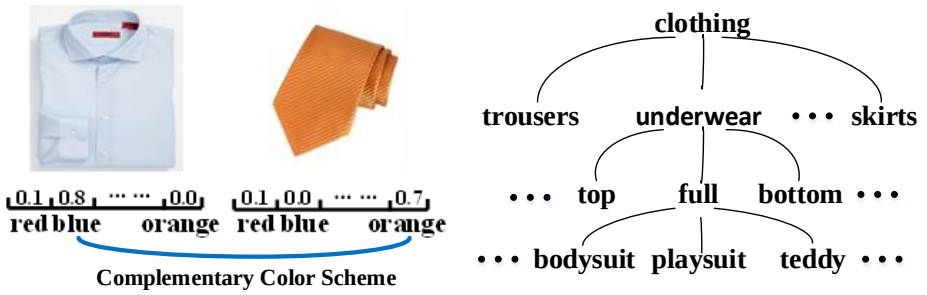
Figure 5.3: Clothing examples of different color schemes.

Color Harmony

Color harmony occurs when two or more different colors are sensed together as a single, pleasing, collective impression [22]. A harmonious piece of outfit shall have pleasing visual effect as compared to one that has not been so carefully put together. For example, a blue T-shirt goes well with an orange tie as shown in Figure 5.4(a). Such color harmony is usually achieved by various color schemes in terms of logical combinations of colors on a color wheel. In this work, we include six color schemes as shown in Figure 5.3: *achromatic scheme, monochromatic scheme, analogous scheme, complementary scheme, split complementary scheme* and *triadic scheme* [41; 135]. These color schemes, establishing the rules of color combinations, help us to identify similar or matchable pieces of clothing.

We combine 10 colors from clothing consultants industry [116] and 12 colors from the classical Prang color wheel [17] [3] in visual Art, to form a total of 14 colors. The 14 colors are: *red, blue, yellow, green, orange, violet, black, white, yellow-orange, red-orange, red-violet, blue-violet, blue-green* and *yellow-green*. Among these colors, *white* and *black* are achromatic colors. In particular, *black* and *white* have not only been known for a long time to combine well with almost any other colors; they have also been widely seen as a pair of complementary

[3] https://cs.nyu.edu/courses/fall02/V22.0380-001/color_theory.htm



(a) Collocation examples (b) Clothing ontology

Figure 5.4: (a) Collocation examples: blue dress shirt goes great with orange tie, as they are complementary colors. (b) An illustrative clothing ontology.

colors. To represent harmonious colors, different from the traditional color histogram representation, we adopt the color naming model of Liu *et.al* [77] by proposing a similar color naming model. The model uses Hue-Saturation-Value(HSV) to map the colors of an image v_i into $\mathbf{p}_i = [c_{i,1}, ..., c_{i,14}]$, where $c_{i,j}$ $(j = 1, ..., 14)$ corresponds to the j-th color of the above 14 ones. The extraction of color features is detailed in Section 5.4.1.

To measure similarity between two images p_i and p_j, we employ *dice (Sorenson's) coefficient* [33] as:

$$d_{sim}(\mathbf{p}_i, \mathbf{p}_j) = \frac{2 \mid \mathbf{p}_i \cap \mathbf{q}_j \mid}{\mid \mathbf{p}_i \mid + \mid \mathbf{q}_i \mid} \tag{5.5}$$

where $\mid \cdot \mid$ means the sum of each element in the vector and $\cap$ means the intersection of two vectors.

If two images are matchable, the colors from separate images, but in the same color scheme would match in a high probability. For example, as shown in Figure 5.4(a) [4], blue t-shirt and orange tie are greatly matchable with high consistent probability of blue (0.8) and orange (0.7) in the same color scheme. Triggered by this, we measure the collocation between pairwise images as:

$$d_{scheme}(\mathbf{p}_i, \mathbf{p}_j) = \sum_{\substack{S \in O \\ m,n \in S \; m \neq n}} c_{i,m} * c_{j,n} \tag{5.6}$$

where O is the set of harmony color schemes shown in Figure 5.3 while S refers to a color scheme set.

[4] http://attireclub.org/2014/05/05/coordinating-the-colors-of-your-clothes/

Note that we have the above two types of distance metrics to measure the degree of matching between two images. We normalize d_{scheme} and d_{sim} into the range of $[0, 1]$. Since this is the first study of applying such color theory to clothing recommendation, we simply measure the matching value d_{ij} of a pair of images as:

$$d_{ij} = \frac{1}{2}(d_{scheme}(\mathbf{p}_i, \mathbf{p}_j) + d_{sim}(\mathbf{p}_i, \mathbf{p}_j)) \tag{5.7}$$

If two images are matchable, their distance in the latent space should be small; and vice vera otherwise. We deploy the following function to measure the weight value z_{ij} between a pair of images based on d_{ij}:

$$\mathbf{Z}_{ij} = \begin{cases} f(d_{ij}) + 1, & \text{if } f(d_{ij}) > 0 \\ f(d_{ij}) - 1, & \text{otherwise.} \end{cases} \tag{5.8}$$

where $f(d_{ij}) = 1/(1 + e^{-(d_{ij}-0.5)*6}) - 0.5$ is a sigmoid function. This enables the corresponding component of $\mathbf{Z}$ in Eq. 5.13 to penalize the distance between matched image pair; and vice vera.

However, such large scale pairwise distance computation among images, would be very insufficient since the item-item matrix would be very large. To speed up the computation, we only consider pairwise images that are very similar or dissimilar in terms of color harmony. We formalize the idea by incorporating only image pair (v_i, v_j) that satisfies the following conditions:

$$\max_{i,j} (d_{scheme}(\mathbf{p}_i, \mathbf{p}_j), d_{sim}(\mathbf{p}_i, \mathbf{p}_j)) < \rho_1 \tag{5.9}$$

$$\max_{i,j} (d_{scheme}(\mathbf{p}_i, \mathbf{p}_j), d_{sim}(\mathbf{p}_i, \mathbf{p}_j)) > \rho_2 \tag{5.10}$$

Following this constraint, we can control the computation speed by setting parameter ρ_1 and ρ_2. In our experiment, we set $\rho_1 = 0.2$ and $\rho_2 = 0.9$, which work well for the experimental datasets.

From color harmony analysis, we finally arrive at a set $\mathbf{\Gamma}$ of image pairs

which includes the set of most likely similar image pairs and dissimilar image pairs. In our work, we separately maintain about 1.0% of similar and dissimilar images with respect to each image for evaluation. This has been found to work well in our experiments.

Clothing Ontology

The hierarchical clothing ontology Λ provides the relationship between different concepts in apparel domain. Human often agrees on the relative relatedness of concepts [90]. For example, most people would agree that *bird* is more related to *feather* than it is to *fork* or *car*. Integrating such semantic relatedness of concepts would faciliate the task of user profiling. Indeed, the concepts in the clothing hierarchy is a subset of keywords $\mathcal{T}$ extracted from images' associated comments and source links. In this work, we employ the idea that the more information two concepts share, the more similar they are in a hierarchy [139], and apply the depth-based similarity measure as follows:

$$\mathbf{O}_{ij} = \frac{2 * depth(LCS(c_i, c_j))}{depth(c_i) + depth(c_j)} \tag{5.11}$$

where c_i and c_j are the concepts in the hierarchy; $LCS(c_i, c_j)$ is the lowest super-ordinate of c_i and c_j; function $depth$ is the depth of a concept in a hierarchy. If two concepts are semantically similar, they would be closer in the learnt latent space. Hence, we penalize the distance between a pair of similar concepts as

$$\mathbf{O}_{ij} \parallel \mathbf{H}_{c_i} - \mathbf{H}_{c_j} \parallel_2^2 \tag{5.12}$$

we employ the clothing ontology from Wikipedia's template such as clothing template [5] and footwear template [6] shown in Figure 5.4(b). The number of concepts and the depth in this hierarchy are 144 and 4 respectively. The extraction of clothing ontology will be detailed in the experiment section 5.4.1.

[5] https://en.wikipedia.org/wiki/Template:Clothing
[6] https://en.wikipedia.org/wiki/Template:Footwear

5.3.3 Unified Model

Model Descriptions

In our unified model, we utilize the aforementioned collective co-factorization with domain knowledge and formalize it as the following objective:

$$\begin{aligned}\min_{\mathbf{W},\mathbf{H},\mathbf{R}} J = &\frac{1}{2}\| \mathbf{M} - \mathbf{X}\mathbf{W}\mathbf{R}^T \|_F^2 + \frac{\alpha}{2}\| \mathbf{Q} - \mathbf{X}\mathbf{W}\mathbf{H}^T \|_F^2 \\ &+ \frac{\beta}{2}\sum_{(v_i,v_j)\in\Gamma} \mathbf{Z}_{ij} \| \mathbf{X}_i\mathbf{W} - \mathbf{X}_j\mathbf{W} \|_2^2 \\ &+ \frac{\gamma}{2}\sum_{(c_i,c_j)\in\Lambda} \mathbf{O}_{ij} \| \mathbf{H}_{c_i} - \mathbf{H}_{c_j} \|_2^2 \\ &+ \frac{\lambda}{2}(\| \mathbf{W} \|_F^2 + \| \mathbf{R} \|_F^2 + \| \mathbf{H} \|_F^2)\end{aligned} \tag{5.13}$$

where α, β, γ and λ are weights to control the tradeoff between different components.

Optimization

The objective function defined in Eq. 5.13 is not convex with respect to the three variables $\mathbf{W}$, $\mathbf{H}$ and $\mathbf{R}$ together. There is no closed-form solution for the problem. Motivated by the multiplicative and alternating updating rules discussed in [74], we now introduce an alternative algorithm to find optimal solutions for the three variables $\mathbf{W}$, $\mathbf{H}$ and $\mathbf{R}$, separately. In particular, we optimize one variable while fixing the others in each iteration. Now we introduce the updating rules in detail.

First we compute $\mathbf{R}$ with $\mathbf{W}$ and $\mathbf{H}$ fixed. When $\mathbf{W}$ and $\mathbf{H}$ are fixed, the corresponding derivative of $\mathbf{R}$ is as follows,

$$\frac{\partial J}{\partial \mathbf{R}} = -\mathbf{M}^T\mathbf{X}\mathbf{W} + \mathbf{R}\mathbf{W}^T\mathbf{X}^\mathbf{T}\mathbf{X}\mathbf{W} + \lambda\mathbf{R} \tag{5.14}$$

Second, we compute $\mathbf{H}$ with $\mathbf{W}$ and $\mathbf{R}$ fixed. When $\mathbf{W}$ and $\mathbf{R}$ are fixed, we compute the derivative of $\mathbf{H}$ as follows,

$$\begin{aligned}\frac{\partial J}{\partial \mathbf{H}} = &\alpha(-\mathbf{Q}^T\mathbf{X}\mathbf{W} + \mathbf{H}\mathbf{W}^T\mathbf{X}^\mathbf{T}\mathbf{X}\mathbf{W}) \\ &+ \gamma\sum_{(c_i,c_j)\in\Lambda} \mathbf{O}_{ij}\frac{\partial \| \mathbf{H}_i - \mathbf{H}_j \|^2}{\partial \mathbf{H}} + \lambda\mathbf{H}\end{aligned} \tag{5.15}$$

where $\frac{\partial \|\mathbf{H}_i - \mathbf{H}_j\|^2}{\partial \mathbf{H}} = \begin{pmatrix} \dots \\ \mathbf{H}_{c_i} - \mathbf{H}_{c_j} \\ \dots \\ \mathbf{H}_{c_j} - \mathbf{H}_{c_i} \end{pmatrix}$. Which means, the c_i row of $\mathbf{H}$ according to $\mathbf{H}_{c_i} - \mathbf{H}_{c_j}$ while the c_j row of $\mathbf{H}$ according to $\mathbf{H}_{c_j} - \mathbf{H}_{c_i}$ for each pair of concepts (c_i, c_j).

Finally, we compute $\mathbf{W}$ with $\mathbf{H}$ and $\mathbf{R}$ fixed. Taking the derivative of J with respect to $\mathbf{W}$, we have,

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{W}} = & (-\mathbf{X}^T\mathbf{M}\mathbf{R} + \mathbf{X}^T\mathbf{X}\mathbf{W}\mathbf{R}^T\mathbf{R}) \\ & + \alpha(-\mathbf{X}^T\mathbf{Q}\mathbf{H} + \mathbf{X}^T\mathbf{X}\mathbf{W}\mathbf{H}^T\mathbf{H}) \\ & + \beta \sum_{(v_i, v_j) \in \Gamma} \mathbf{Z}_{ij} \, (\mathbf{X}_i - \mathbf{X}_j)^T (\mathbf{X}_i - \mathbf{X}_j)\mathbf{W} + \lambda \mathbf{W} \end{aligned} \tag{5.16}$$

The widely used mini-batch stochastic gradient descent (SGD) is adopted to optimize each variable. The main process of mini-batch SGD on alternative optimization is that when updating each variable, a) the training dataset is randomly shuffled into batches in each iteration; b) based on each batch's training samples, the variable is updated according to the above equations; c) for each iteration, if it converges, then we update next variable. We keep this procedure until the training objective function converges. In our experiment, we stop the iterations either when the improvement in training error is smaller than some threshold (0.001) or when we reach the maximum number of iterations.

5.4 Experiments

In this section, we introduce experimental details to validate the effectiveness of the proposed method. We first introduce our experimental setup. We then provide the parameter analysis followed by comparative performances with variants of the proposed model and existing state-of-the-art approaches. Finally, we provide an extension study to show the scalability of the proposed model.

Table 5.2: Statistics of Pinterest dataset

Number of users	12,809
Number of boards	13,397
Number of pins	2,275,712
Number of images	956,278
Number of source links from images	864,725
Number of descriptions from pins	2,023,845
Sharing Duration	Aug, 2014 - July, 2015
Sparsity	99.98%

Table 5.3: Statistics of Amazon dataset

Number of users	1,334
Number of images	52,845
Number of reviews	62,014
Purchasing Duration	Aug, 2010 - July, 2014
Sparsity	99.92%

5.4.1 Experimental Setup

Datasets

Recall that our proposed model is general and can be utilized for different scenarios. Hence, in this paper, we evaluate our methods on two datasets: one is from content-centric network, Pinterest [7] ; and the other from Amazon [8].

Overall, the data acquisition contains the following steps. First, we browsed the category of *women-fashion* [9] in Pinterest to crawl a set of fashion related pins; and manually selected a set of seeds from these pins. Second, we crawled the boards that own these pins from the seed. Simultaneously, we crawled the boards that repin these pins from the seed.

The second dataset we employed for evaluation comes from Amazon [84]. This dataset consists of product images, product categories, product co-purchase information and product reviews information. Note that while the dataset contains products from diverse categories, we only consider the "Clothing, Shoes and Jewelry" category and its subcategories since this work takes apparel domain as an example.

[7] https://www.pinterest.com/
[8] http://www.amazon.com/
[9] https://www.pinterest.com/categories/womens_fashion/

Ground Truths

We preprocessed the datasets from Pinterest and Amazon to obtain ground truths for evaluation.

For Pinterest, we first removed boards that have less than 10 pins or more than 800 pins according to the power law pin-board distribution. Second, we selected one year of user data for evaluation: we used data from August of 2014 to May of 2015 as training; and data in the following two months as testing. In this process, we involved only users who have more than 10 pins in the training set and removed those users who do not have any sharing in the testing set. Finally, we obtained a large scale dataset as shown in Table 5.2.

To prepare the Amazon dataset, we selected 4 years of users' purchase history: we used data from August of 2010 to January of 2014 as training; and data in the remaining half-a-year as testing. In this process, we involved only users who have more than twenty purchasing records in the training set and removed users who do not have any purchases in the testing set. Table 5.3 shows the statistics of the final Amazon dataset.

During training, the training set is used to train the user profile model. During testing, we generate a ranked list of images from the testing set for each user based on his/her profile, and measure the image recommendation performance. Our proposed model can be used for both in-matrix and out-matrix recommendation tasks. As discussed in [132], in-matrix recommendation refers to the case where the user has not rated an item but that item has been rated by at least one other user; while out-matrix recommendation refers to the case where none of the users have rated a particular item, *i.e.* the item has no rating records. We conducted these two types of evaluation in our experiments.

Feature Extraction

Image Content Feature Extraction To remove background noise and to focus on the clothing parts of images, we adopted the state-of-the-art Fast R-CNN [48]

model on PASCAL VOC 2007 to detect person. We set the default detection threshold as 0.8 [48] and selected the most probable box containing person to crop images. We then deployed the 22-layer deep convolutional network GoogLeNet [122], a top-performing entry of the ILSVRC-2014 classification task, to extract $1,024$-D deep features for images on a NVIDIA Titian Z GPU in the 64G memory machine.

Keyword Extraction Images on Pinterest have comments, descriptions and source links. For image source web pages, we need to extract textual contents that are relevant to an image while removing irrelevant information. There are several places where relevant text may be found, namely, (a) external source url; (b) page title; and (c) page description in the header of source html [43] that tend to provide the most accurate description of the embedded image. Our keyword set comes from two components. One is from Wikipedia clothing ontology. Through conducting template mapping of associated texts to the concepts of Wikipedia clothing ontology, useful concepts occurred in associated texts were extracted. Another is from the frequently occuring unigrams and bigrams in image associated texts. After conducting stemming and stopwords removing, we selected NLTK [20] as chunk parsing tools to extract noun phrases and selected the most frequent unigrams and bigrams from these extracted noun phrases. Lastly, we obtained $2,858$ keywords for Pinterest and $1,269$ keywords for Amazon separately.

Color Harmony Feature Extraction For color histograms based on color harmony, we uniformly quantize the Hue value to determine 12 base colors from Prang color wheel. For *black* and *white* colors, when $saturation < 0.1$ and $value > 0.8$, we get the "white" color and when $value < 0.1$, we always get the "black" color [77].

Comparison Methods

To investigate the effectiveness and the efficiency of the proposed approach on image recommendation, we selected several recent representative methods for comparison:

- **Random(Rand)**: Candidate images are randomly selected from in-matrix image set and out-matrix image set.
- **Most Popular(MP)**: This method presents a non-personalized ranked item list based on the popularity of items among all users.
- **MBCF**: This memory-based approach [7; 15] deploys an asymmetric similarity measure between user-based collaborative filtering and item-based collaborative filtering to mine users' positive binary feedback for recommendation.
- **CLiMF**: This method [108] presents a collaborative-filtering algorithm able to directly maximize the mean reciprocal rank (MRR) of relevant items instead of trying to predict ratings.
- **CTR**: This approach [132] combines the traditional collaborative filtering with probabilistic topic modeling, that results in latent features for users and items. Then, items are recommended by the similarity between the user features and the item features.
- **LCE**: This method [107] exploits user-item matrix and item-feature matrix from past user behaviors and items' properties while enforces the manifold structure exhibited by the collective embedding, that are the learnt user and item features.

We denote our solution as **EmLUCK**. We are also interested in the effectiveness of different components in our proposed model. In particular, we compared the performance of incorporating image-keyword associations(**K**), color harmony(**C**) and clothing ontology(**O**) separately. We hence conducted experiments to comparatively validate the experimental settings as shown in Table 5.4.

Table 5.4: Proposed approach and its variants

Approaches	image-user matrix (**U**)	image-keyword matrix (**K**)	color harmony (**C**)	clothing ontology (**O**)
EmLUCK-KCO	✓	-	-	-
EmLUCK-CO	✓	✓	-	-
EmLUCK-KO	✓	-	✓	-
EmLUCK-C	✓	✓	-	✓
EmLUCK	✓	✓	✓	✓

In our experiments, we compare our model with the comparing methods on in-matrix image recommendation and out-matrix image recommendation separately. As far as we know, except for **Random**, **LCE** and **CTR**, all the other comparing methods are only able to conduct in-matrix image recommendation. For all the baselines, we tuned their parameters involved based on the methods provided in the respective papers and selected the best values.

Evaluation Metrics

In this work, we adopted three popular metrics of Recall@k, Precision@k and Normalized Discounted Cumulative Gain (NDCG), to measure the effectiveness of the proposed approach.

Given a set of recommended relations of a given type *rec*, and a set of known-relevant products *rel*, the precision is defined as

$$precision = |rel \cap rec|/|rec| \tag{5.17}$$

i.e., the fraction of recommended items that are relevant. While recall is defined as

$$recall = |rel \cap rec|/|rel| \tag{5.18}$$

The Precision@k and Recall@k is then the precision obtained given a fixed budget, *i.e.*, when $|rec| = k$.

Normalized Discounted Cumulative Gain (NDCG) measures the recommendation performance of a recommendation system based on the graded relevance

Table 5.5: Performance comparisons of variants of the proposed model for existing image recommendation on Amazon and Pinterest datasets.

Methods	Amazon			Pinterest		
	R@10	P@10	NDCG	R@10	P@10	NDCG
EmLUCK-KCO	0.71%	0.19%	0.49%	0.31%	0.12%	0.23%
EmLUCK-KO	0.80%	0.24%	0.59%	0.38%	0.17%	0.29%
EmLUCK-CO	0.95%	0.27%	0.67%	0.43%	0.21%	0.31%
EmLUCK-C	0.98%	0.29%	0.71%	0.47%	0.24%	0.35%
EmLUCK	**1.12%**	**0.31%**	**0.78%**	**0.55%**	**0.26%**	**0.43%**

Table 5.6: Performance comparisons of variants of the proposed model for cold-start image recommendation on Amazon and Pinterest datasets.

Methods	Amazon			Pinterest		
	R@10	P@10	NDCG	R@10	P@10	NDCG
EmLUCK-KCO	1.97%	2.11%	2.67%	0.72%	0.34%	0.61%
EmLUCK-KO	2.01%	2.22%	2.84%	0.73%	0.36%	0.65%
EmLUCK-CO	2.03%	2.35%	2.91%	0.72%	0.35%	0.63%
EmLUCK-C	2.02%	2.31%	2.92%	0.72%	0.35%	0.64%
EmLUCK	**2.26%**	**2.51%**	**3.14%**	**0.75%**	**0.39%**	**0.67%**

of the recommended entities.

$$NDCG_k = \frac{1}{IDCG_k} \times \sum_{i=1}^{k} \frac{2^{r_i - 1}}{log_2(i+1)} \tag{5.19}$$

where $IDCG_k$ is the maximum possible(ideal) $NDCG_k$ and r_i is the degree of relevance of the image in position i. In our experiment, we deployed Recall@10(R@10), Precision@10(P@10) and NDCG@10(NDCG) as evaluation metrics.

We have tried different configurations and finally set the batch size of mini-SGD to be $10,000$. The final parameter settings we used are: $\alpha = 0.01$, $\beta = 0.001$, $\gamma = 0.01$, $\lambda = 1.0$, learning rate $lr = 0.0001$, and latent dimension $l = 600$. Besides, we empirically set $\rho_1 = 0.9$ and $\rho_2 = 0.2$ in Eq. 5.9 and Eq. 5.10, respectively.

5.4.2 Effects of Components

Table 5.5 and Table 5.6 respectively present the results of in-matrix recommendation and out-matrix recommendation for different variants of our proposed model on the Amazon and Pinterest datasets in terms of Recall@10,

Precision@10 and NDCG@10. The tables shows that the more components we incorporate, the better the performance can be achieved. This indicates the comparatively complementary relationships instead of mutual conflicting relationships among the different components in recommendation.

From Table 5.5, we noted that the performance improvements obtained from different components are not the same. Interestingly, we noted that **EmLUCK-CO** achieves a better performance as compared to **EmLUCK-KCO**, which does not use ontology. This may be due to the existence of a high proportion of image-keyword associations in the datasets and that most keywords have good semantic meanings in the apparel domain and hence the effects of noisy keywords are reduced. Therefore, the co-factorization of image-keyword associations and image-user associations can help to improve the performance. Second, the combination of image-user associations and color harmony (**EmLUCK-KO**) has achieved 12.68% improvement in terms of Recall@10 as compared to the single image-user association (**EmLUCK-KCO**). This is a very promising result as it indicates that the color harmony can bring in useful similar and dissimilar image pairs to improve the performance of the proposed model. This can also be observed from the comparison between method **EmLUCK-C** and **EmLUCK**. Third, **EmLUCK-C** improved the performance by 9.30% as compared to **EmLUCK-CO** on Pinterest dataset while only 3.16% on Amazon dataset. The result indicates the importance of ontology and the relatedness between concepts, which is higher on Pinterest dataset. This is because Pinterest has more text meta-data from images' descriptions and source links and these texts also have a higher ratio of clothing concepts as compared to that in Amazon. Such a higher ratio of clothing concepts also highlights the role of ontology and helps in achieving better performance.

Although we can also see that the combination of all components performs the best in Table 5.6, however, as compared to Table 5.5, each component plays comparatively less role in out-matrix image recommendation. This is attributed

Table 5.7: Performance comparisons of baselines for existing image recommendation on Amazon and Pinterest datasets.

Methods	Amazon			Pinterest		
	R@10	P@10	NDCG	R@10	P@10	NDCG
Rand	0.01%	0.01%	0.01%	0.004%	0.016%	0.012%
MP	0.64%	0.28%	0.24%	0.14%	**0.26%**	0.30%
MBCF	0.23%	0.15%	0.19%	0.09%	0.05%	0.10%
CLiMF	0.76%	0.23%	0.69%	0.17%	0.12%	0.19%
LCE	0.16%	0.11%	0.14%	0.07%	0.05%	0.08%
CTR	0.50%	0.23%	0.67%	0.21%	0.15%	0.24%
EmLUCK	**1.12%**	**0.31%**	**0.78%**	**0.55%**	**0.26%**	**0.43%**

Table 5.8: Performance comparisons of baselines for cold-start image recommendation on Amazon and Pinterest datasets.

Methods	Amazon			Pinterest		
	R@10	P@10	NDCG	R@10	P@10	NDCG
Rand	0.05%	0.02%	0.02%	0.028%	0.091%	0.011%
LCE	0.48%	0.25%	0.32%	0.11%	0.07%	0.14%
CTR	1.16%	1.31%	1.54%	0.54%	0.30%	0.19%
EmLUCK	**2.26%**	**2.51%**	**3.14%**	**0.75%**	**0.39%**	**0.67%**

to the fact that cold-start images contain no associated texts, and have no pre-computed similar/dissimilar images according to color harmony. However, we can see that the absolute performance of out-matrix recommendation is higher than that of in-matrix recommendation. This is misleading because the number of relevant images in the dataset for the out-matrix cases is much lower than that for the in-matrix cases and hence the chances of recommending the correct images for each user is higher. Note that in our evaluation, we considered all relevant images selected by all users during the testing phase to be relevant and used that to evaluate a user's actual selection. This evaluation is very strict and tends to give very low value when the number of relevant images is very high.

5.4.3 Performance Comparisons with State-of-the-Art Approaches

We now compare the performance of our approach with the baselines as listed in Section 5.4.1. Table 5.7 shows the comparative performance of in-matrix recommendation; while Table 5.8 details that of the out-matrix recommendation

on the two datasets. From the results, we can see that our approach shows significant improvements as compared to the other baseline algorithms on both in-matrix and out-matrix recommendation. First, the results show a powerful capacity of **EmLUCK** on recommendation performance as compared to other matrix factorization approaches *i.e.*, **MBCF** and **CLiMF**. This is due to **EmLUCK**'s ability to in handle data sparsity and the use of content features. It is noted that the user-image associations are highly sparse and such sparsity can affect the performance of different approaches. As can be seen from Table 5.2 and Table 5.3, Pinterest is much sparser than the Amazon dataset. As a result, we can see that **MBCF** and **CLiMF** have much lower performance on Pinterest dataset as compared to Amazon dataset; whereas that for **EmLUCK** are less but with much higher absolute performance. Besides, both **MBCF** and **CLiMF** do not incorporate items' contents which further limit their performance. Second, when comparing to **LCE** and **CTR**, that consider only items' contents, **EmLUCK** also displays its superiority on image recommendation due to the use of additional image-keyword associations. The co-factorization of image-user associations and image-keyword associations could lead to better performance as also can be seen from the performance of **EmLUCK-CO** as compared to the other baselines. This result is consistent with the observation in [111] that mixing information from multiple relations leads to better performance. Third, we also observed that the use of two kinds of human prior knowledge, namely, color harmony and clothing ontology, lead to improved performance. Finally, our experiments found that the simple approach **MP** shows a comparatively good performance as compared to other baselines. One explanation could be that many commercial sites conduct recommendation to users based partially on the popularity of products.

Amazon				
jewelry	sunglasses	watches	bras	pants
silver	glass	watch	bra	pant
jewelry stone	sunglass	movement	cup	tight
sterling	frame	date	breast coverage	leg
sparkle gift	eye	wrist	shape	stretch hip
shine	len	bezel	cup size	thigh
diamond earring	sun	invicta band	woman	jean
delicate	shade	swiss	sexy	cut
tiny	protection	citizen seiko	hook	denim
earring	nose	great watch	sports bra	fitting length
affordable	cloth	quartz	cleavage	low waist
Pinterest				
dress	bags	street style	shoes	makeup
wedding dress	handbag	summer street	heel	nail design
dress	bag chanel	street fashion	sandal	eye makeup
wedding gown	bucket	outfit	naked	nail art
sleeveless	clutch	casual style	perfect shoes	wedding hairstyle
sheath dress	handbag	style trend	zipper	haircut inspiration
pink	kor bag	spring	flat sandal	lips makeup
party	leather bag	fashion week	pump	Necklace
red	lv	fashion	ankle strap	eyeshadow
woman dress	shoulder	fall outfit	lattice	haircolor
maxi	messenger	street chic	louboutin	pink lips

Table 5.9: Top ten keywords from selected topics discovered in Amazon and Pinterest. Each column is labeled with an "interpretation" of that topic.

5.4.4 Extension Study

Interestingly, our model is able to learn the latent vectors of keywords. We conduct topic detections in Amazon and Pinterest separately by clustering the learnt topics. Some of the topics discovered by our model are shown in Table 5.9. For example, our model has the ability to detect some brands such as "seiko" and "swiss" of "watches". This result is consistent with previous study [83] on topic modeling. This demonstrates that the proposed model not only can learn the latent space of users and images for recommendation, but also latent topics in semantic space.

In addition to topic modeling, our model can be used in different applications such as friend recommendation and community detection by measuring the similarity among users. Moreover, by clustering the learnt embedding of users, images and contents, a comprehensive user profile can be constructed from a latent-based and interpretable way as shown in Figure 5.1. What's more, the feature obtained from color harmony of visual Art plays an important role in our recommendation task. Such feature can be seen as a new kind of feature to measure the similarity between different pieces of clothing. It is reasonable that many users may like certain specific combinations of colorful clothes. Since traditional approaches only consider the content similarity without matchable measurement, such an approach would help to identify more meaningful pieces of clothes as shown in Figure 5.3.

It is worth noting that the incorporation of prior knowledge is not restricted to apparel domain. Such knowledge-guided embedding approach with heterogenous networks of users and contents can be generalized to other domains if the corresponding domain knowledge exist.

5.5 Summary

Recommendation has become an important element in almost all kinds of online commercial sites. The distinct characteristics of social media such as diverse multimedia contents and sparse user-item associations present new challenges for recommendation in social media. Driven by the desire to obtain semantic and efficient user profiles, in this paper, we emphasized the heterogenous information of users and contents to learn a latent space. Additionally, we utilized two kinds of human prior knowledge, namely, color harmony and clothing ontology, to guide the representation learning. Experimental results on two real-world datasets demonstrate the importance of color harmony and heterogenous user-content connections. Based on the learnt embedding, we can easily infer the semantic and visual aspects of users' interests, leading to many applications such as advertisement targeting and commercial recommendation. Moreover, the learnt embedding of users, images and texts are also useful in different applications such as topic detection and community detection.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

This thesis focuses on the task of user profiling based on rich media interactions in OSNs. Considering that user profiling can be inferred from the rich media content analysis, this thesis proposed two approaches: user profiling by knowledge-based multi-task learning, and user profiling by deep learning of user-media interactions, respectively. Moreover, this thesis practically applied the proposed approaches to personalized recommendation.

The first approach performs the knowledge-based multi-task learning to enhance media understanding by exploiting the intuition that sibling nodes share common visual attributes in a hierarchical ontology while are exclusive in the fine-grained detailed visual information. The proposed approach is able to automatically construct profile ontology and then jointly learn features of a node concept and its siblings. To improve the profiling results, it also proposed a low-rank recovery framework to further refine the generated user profiles by the ontological profile models, exploiting the rich user-level, bundle-level and content-level social relations offered by social curation. The experimental results enable us to draw the following conclusions. First, the rich media content analysis does improve the user profiling performance. This demonstrates that for

multimedia-based OSNs, it is essential to incorporate user-media interactions aside from purely textual information. Second, utilizing knowledge to guide the media feature learning arguments the performance of user profiling. This demonstrates that human knowledge can improve the performance of state-of-the-art media understanding approaches. Third, the rich social connections, as one kind of social collective intelligence in OSNs, can help to boost the user profiling performance.

The second approach performs deep learning of user-media interactions to mine the rich user-media connections to enhance feature learning of media and users simultaneously. The proposed approach breaks a large and sparse network topology into a tree-structured deep hierarchy, where the leafs are users and images. The model can compactly and efficiently learn representative features to reveal the weak correlations between images and users at the scene of the extremely sparse connections and extremely diverse images due to its deep structure. Specifically, we made use of the specific observation in social media that the connections between users and contents are very sparse and then introduced a synchronization optimization algorithm to ensure a fast and accurate learning process. The learnt representative low-dimensional vectors of users and images can be directly applied to many applications such as personalized recommendation and community detection. The experimental results show that as compared with state-of-the-art content-based user profiling or collaborative filtering user profiling approaches, jointly analyzing image contents and social curation network topology can boost the performance of image understanding and user profiling significantly.

The third approach performs a matrix co-factorization on the heterogenous networks of contents and users, guided by the human prior knowledge. Specifically, it is able to map users and the multimedia contents they shared into a compact common space. Consequently, the recommendation of images to users can be conducted by directly measuring the similarity between users

and images; and friend recommendation can also be done in a similar way. Moreover, by measuring the similarity between users and texts, a semantic-based user profile can also be constructed. The experimental results show that as compared with state-of-the-art recommendation approaches, collaboratively learning features for images, users and texts in the heterogenous network can improve the performance for user profiling and recommendation significantly. Besides, domain knowledge such as clothing ontology and color harmony can help to guide the feature learning in the above process.

6.2 Future Directions

There are a few interesting extensions towards more accurate and comprehensive multimedia user profiling in social media.

First, in the current work, we have not considered integrating the domain knowledge directly into the rich media feature extraction. This could advance the performance of feature learning in different multimedia applications. Note that this is challenging since different domains have their own intrinsic knowledge. Furthermore, with the diverse characteristics of different domains, how to develop a user profiling approach that performs universally well in all image domains is also quite challenging.

Second, user preferences are often time-sensitive. For example, the emergence of new products or services often changes the focus of customers. Related to this are seasonal changes, or specific holidays, which lead to characteristic shopping patterns. Due to various reasons, users' interest often change suddenly or smoothly. This process is highly complex, since for different customers, different types of concept drifts may exist and each concept drift may occur at a distinct time frame and is driven towards a different direction [67]. Therefore, it is essential to find an alternative way to learn user preferences by taking into account the temporal information. In the future, it would be a quite

interesting direction to combine the temporal sequential information with rich media analysis to extract user interests.

Third, in the era of Web 2.0, users are often active in a number of social networks for different purposes. For example, users may connect with their business partners via LinkedIn while they may connect with their family members and friends in Facebook to update their personal experience. It could be an interesting research direction to develop strategies to bring the multimedia contents distributed in different social networks towards more comprehensive and accurate user profiling for many personalized services.

This thesis focuses more on the effective multimedia user profiles in OSNs and hence the proposed approaches have limitations on text-based sites such as Twitter. As this thesis aims to address the issue of the fundamental task of personalization, *i.e.*, user profiling, we believe that our approaches are not limited to image recommendation, but can be applied to other applications such as community detection, image annotation and topic detection. Moreover, even though this work is carried out based on the recommendation in Pinterest, we believe that our proposed methods can be robustly extended to other multimedia-based OSNs.

Bibliography

[1] R. Abbasi, S. Chernov, W. Nejdl, R. Paiu, and S. Staab. Exploiting flickr tags and groups for finding landmark photos. In *ECIR*. Springer, 2009.

[2] F. Abel, Q. Gao, G.-J. Houben, and K. Tao. Analyzing user modeling on twitter for personalized news recommendations. In *UMAP*. Springer, 2011.

[3] B. Adams, D. Phung, and S. Venkatesh. Extraction of social context and application to personal multimedia exploration. In *ACM SIGMM*, 2006.

[4] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *TKDE*, 2005.

[5] D. Agarwal and B.-C. Chen. Regression-based latent factor models. In *ACM SIGKDD*, 2009.

[6] T. Ahonen, A. Hadid, and M. Pietikäinen. Face recognition with local binary patterns. In *ECCV*, 2004.

[7] F. Aiolli. Efficient top-n recommendation for very large scale binary rated datasets. In *ACM RecSys*, 2013.

[8] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanhalli. Multimodal fusion for multimedia analysis: a survey. *Multimedia systems*, 2010.

[9] M. Balabanović and Y. Shoham. Fab: content-based, collaborative recommendation. *Communications of the ACM*, 1997.

[10] S. Banerjee, K. Ramanathan, and A. Gupta. Clustering short texts using wikipedia. In *ACM SIGIR*, 2007.

[11] B. Bazelli, A. Hindle, and E. Stroulia. On the personality traits of stackoverflow users. In *ICSM*, 2013.

[12] Y. Bengio. Learning deep architectures for ai. *Foundations and trends in Machine Learning*, 2009.

[13] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *TPAMI*, 2013.

[14] Y. Bengio, H. Schwenk, J.-S. Senécal, F. Morin, and J.-L. Gauvain. Neural probabilistic language models. In *Innovations in Machine Learning*. Springer, 2006.

[15] D. Bernardes, M. Diaby, R. Fournier, F. FogelmanSoulié, and E. Viennet. A social formalism and survey for recommender systems. *ACM SIGKDD Explorations Newsletter*, 2015.

[16] T. Berners-Lee, J. Hendler, O. Lassila, et al. The semantic web. *Scientific american*, 2001.

[17] W. v. Bezold. The theory of color. *Louis Prang, Boston, Massachusetts*, 1876.

[18] A. Bhardwaj, A. Das Sarma, W. Di, R. Hamid, R. Piramuthu, and N. Sundaresan. Palette power: enabling visual search through colors. In *ACM SIGKDD*, 2013.

[19] B. Bi, M. Shokouhi, M. Kosinski, and T. Graepel. Inferring the demographics of search users: Social data meets search queries. In *ACM WWW*, 2013.

[20] S. Bird. Nltk: the natural language toolkit. In *ACL*, 2006.

[21] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *CVPR*, 2010.

[22] K. E. Burchett. Color harmony. *Color Research & Application*, 2002.

[23] V. Challam, S. Gauch, and A. Chandramouli. Contextual search using ontology-based user profiles. In *Large Scale Semantic Access to Content (Text, Image, Video, and Sound)*, 2007.

[24] S. Chang, W. Han, J. Tang, G.-J. Qi, C. C. Aggarwal, and T. S. Huang. Heterogeneous network embedding via deep architectures. In *ACM SIGKDD*, 2015.

[25] M.-y. Chen and A. Hauptmann. Mosift: Recognizing human actions in surveillance videos. 2009.

[26] W. Chu and S.-T. Park. Personalized recommendation on dynamic content using predictive bilinear models. In *ACM WWW*, 2009.

[27] T.-S. Chua, H. Luan, M. Sun, and S. Yang. Next: Nus-tsinghua center for extreme search of user-generated content. *MultiMedia, IEEE*, 2012.

[28] A. Clauset, M. E. Newman, and C. Moore. Finding community structure in very large networks. *Physical review E*, 2004.

[29] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[30] M. De Choudhury, H. Sundaram, A. John, and D. D. Seligmann. What makes conversations interesting?: themes, participants and consequences of conversations in online social media. In *ACM WWW*, 2009.

[31] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, A. Senior, P. Tucker, K. Yang, Q. V. Le, et al. Large scale distributed deep networks. In *NIPS*, 2012.

[32] M. Deshpande and G. Karypis. Item-based top-n recommendation algorithms. *ACM TOIS*, 2004.

[33] L. R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 1945.

[34] N. Dimitrova, H.-J. Zhang, B. Shahraray, I. Sezan, T. Huang, and A. Zakhor. Applications of video-content analysis and retrieval. *IEEE multimedia*, 2002.

[35] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv preprint arXiv:1310.1531*, 2013.

[36] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014.

[37] M. Efron, P. Organisciak, and K. Fenlon. Improving retrieval of short texts through document expansion. In *ACM SIGIR*, 2012.

[38] M. Eirinaki and M. Vazirgiannis. Web mining for web personalization. *ACM TOIT*, 2003.

[39] T. Evgeniou and M. Pontil. Regularized multi–task learning. In *ACM SIGKDD*, 2004.

[40] A. Farseev, L. Nie, M. Akbari, and T.-S. Chua. Harvesting multiple sources for user profile learning: a big data study. In *ACM ICMR*, 2015.

[41] E. A. Feisner. *Colour: how to use colour in art and design*. Laurence King Publishing, 2006.

[42] R. Feldman and I. Dagan. Knowledge discovery in textual databases (kdt). In *KDD*, 1995.

[43] H. Feng, R. Shi, and T.-S. Chua. A bootstrapping framework for annotating and retrieving www images. In *ACM SIGMM*, 2004.

[44] S. Gauch, J. Chaffee, and A. Pretschner. Ontology-based personalized search and browsing. *Web Intelligence and Agent Systems: An international Journal*, 2003.

[45] S. Gauch, M. Speretta, A. Chandramouli, and A. Micarelli. User profiles for personalized information access. In *The adaptive web*. Springer, 2007.

[46] J.-L. Gauvain and C.-H. Lee. Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *TASLP*, 1994.

[47] T. Gevers and A. W. Smeulders. Pictoseek: Combining color and shape invariant features for image retrieval. *TIP*, 2000.

[48] R. Girshick. Fast r-cnn. In *ICCV*, 2015.

[49] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 2001.

[50] A. Graves, A.-r. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, 2013.

[51] Q. Gu, J. Zhou, and C. H. Ding. Collaborative filtering: Weighted nonnegative matrix factorization incorporating user and item graphs. In *SDM*, 2010.

[52] I. Guy, N. Zwerdling, I. Ronen, D. Carmel, and E. Uziel. Social media recommendation based on people and tags. In *ACM SIGIR*, 2010.

[53] J. Han and K.-K. Ma. Fuzzy color histogram and its use in color image retrieval. *TIP*, 2002.

[54] A. Hanjalic, R. Lienhart, W.-Y. Ma, and J. R. Smith. The holy grail of multimedia information retrieval: So close or yet so far away? *Proceedings of the IEEE*, 2008.

[55] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[56] G. E. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 2006.

[57] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 1997.

[58] X. Hu, N. Sun, C. Zhang, and T.-S. Chua. Exploiting internal and external semantics for the clustering of short texts using world knowledge. In *ACM CIKM*, 2009.

[59] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *ICDM*, 2008.

[60] K. Ishiguro, A. Kimura, and K. Takeuchi. Towards automatic image understanding and mining via social curation. In *ICDM*, 2012.

[61] A. K. Jain and B. Chandrasekaran. 39 dimensionality and sample size considerations in pattern recognition practice. *Handbook of statistics*, 1982.

[62] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM SIGMM*, 2014.

[63] T. Joachims. *Text categorization with support vector machines: Learning with many relevant features*. Springer, 1998.

[64] N. Kalchbrenner and P. Blunsom. Recurrent continuous translation models. In *EMNLP*, 2013.

[65] J. Kalyanam, A. Mantrach, D. Saez-Trumper, H. Vahabi, and G. Lanckriet. Leveraging social context for modeling topic evolution. In *ACM SIGKDD*, 2015.

[66] S.-B. Kim, K.-S. Han, H.-C. Rim, and S. H. Myaeng. Some effective techniques for naive bayes text classification. *TKDE*, 2006.

[67] Y. Koren. Collaborative filtering with temporal dynamics. *Communications of the ACM*, 2010.

[68] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 2009.

[69] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[70] I. Laptev. On space-time interest points. *IJCV*, 2005.

[71] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[72] Q. V. Le. Building high-level features using large scale unsupervised learning. In *ICASSP*, 2013.

[73] Y. LeCun and Y. Bengio. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 1995.

[74] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, 2001.

[75] D. D. Lewis and M. Ringuette. A comparison of two learning algorithms for text categorization. In *Third annual symposium on document analysis and information retrieval*, 1994.

[76] D. Liu, X.-S. Hua, M. Wang, and H.-J. Zhang. Image retagging. In *ACM SIGMM*, 2010.

[77] Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma. Region-based image retrieval with high-level semantic color names. In *MMM 2005*, 2005.

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

[79] B. S. Manjunath and W.-Y. Ma. Texture features for browsing and retrieval of image data. *TPAMI*, 1996.

[80] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*. Cambridge university press Cambridge, 2008.

[81] D. Markovikj, S. Gievska, M. Kosinski, and D. Stillwell. Mining facebook data for predictive personality modeling. In *ICWSM*, 2013.

[82] M. Marszalek and C. Schmid. Semantic hierarchies for visual object recognition. In *CVPR*, 2007.

[83] J. McAuley, R. Pandey, and J. Leskovec. Inferring networks of substitutable and complementary products. In *ACM SIGKDD*, 2015.

[84] J. McAuley, C. Targett, Q. Shi, and A. van den Hengel. Image-based recommendations on styles and substitutes. In *ACM SIGIR*, 2015.

[85] M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual review of sociology*, 2001.

[86] S. E. Middleton, N. R. Shadbolt, and D. C. De Roure. Ontological user profiling in recommender systems. *ACM TOIS*, 2004.

[87] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[88] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[89] T. Mikolov and G. Zweig. Context dependent recurrent neural network language model. *SLT*, 2012.

[90] G. A. Miller and W. G. Charles. Contextual correlates of semantic similarity. *Language and cognitive processes*, 1991.

[91] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel. You are who you know: inferring user profiles in online social networks. In *ACM WSDM*, 2010.

[92] A. Mnih and R. Salakhutdinov. Probabilistic matrix factorization. In *NIPS*, 2007.

[93] F. Morin and Y. Bengio. Hierarchical probabilistic neural network language model. In *AISTATS*, 2005.

[94] M. Naaman. Social multimedia: highlighting opportunities for search and mining of multimedia data in social media applications. *Multimedia Tools and Applications*, 2012.

[95] M. E. Newman. Modularity and community structure in networks. *PNAS*, 2006.

[96] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 2001.

[97] G. Pass, R. Zabih, and J. Miller. Comparing images using color coherence vectors. In *ACM SIGMM*, 1997.

[98] M. J. Pazzani and D. Billsus. Content-based recommendation systems. In *The adaptive web*. Springer, 2007.

[99] B. Perozzi, R. Al-Rfou, and S. Skiena. Deepwalk: Online learning of social representations. In *ACM SIGKDD*, 2014.

[100] N. Qian. On the momentum term in gradient descent learning algorithms. *Neural networks*, 1999.

[101] D. Quercia, R. Lambiotte, D. Stillwell, M. Kosinski, and J. Crowcroft. The personality of popular facebook users. In *CSCW*, 2012.

[102] D. Rao, D. Yarowsky, A. Shreevats, and M. Gupta. Classifying latent user attributes in twitter. In *SMUC*, 2010.

[103] E. Riloff and W. Lehnert. Information extraction as a basis for high-precision text classification. *ACM TOIS*, 1994.

[104] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *ACM WWW*, 2010.

[105] G. Salton, A. Wong, and C.-S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 1975.

[106] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Application of dimensionality reduction in recommender system-a case study. Technical report, DTIC Document, 2000.

[107] M. Saveski and A. Mantrach. Item cold-start recommendations: learning local collective embeddings. In *ACM RecSys*, 2014.

[108] Y. Shi, A. Karatzoglou, L. Baltrunas, M. Larson, N. Oliver, and A. Hanjalic. Climf: learning to maximize reciprocal rank with collaborative less-is-more filtering. In *ACM RecSys*, 2012.

[109] A. Sieg, B. Mobasher, and R. D. Burke. Learning ontology-based user profiles: A semantic approach to personalized web search. *IEEE Intelligent Informatics Bulletin*, 2007.

[110] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[111] A. P. Singh and G. J. Gordon. Relational learning via collective matrix factorization. In *ACM SIGKDD*, 2008.

[112] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, 2003.

[113] J. R. Smith. Riding the multimedia big data wave. In *ACM SIGIR*, 2013.

[114] S. Soderland. Learning information extraction rules for semi-structured and free text. *Machine learning*, 1999.

[115] X. Song, L. Nie, L. Zhang, M. Akbari, and T.-S. Chua. Multiple social network learning and its application in volunteerism tendency prediction. In *ACM SIGIR*, 2015.

[116] M. Spillane. *The Complete Style Guide from the Color Me Beautiful Organization*. Not Applicable, 1991.

[117] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas. Short text classification in twitter to improve information filtering. In *ACM SIGIR*, 2010.

[118] N. Srivastava, R. R. Salakhutdinov, and G. E. Hinton. Modeling documents with deep boltzmann machines. *arXiv preprint arXiv:1309.6865*, 2013.

[119] X. Su and T. M. Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009.

[120] K. Sugiyama, K. Hatano, and M. Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *WWW*, 2004.

[121] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[122] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[123] J. Tang, M. Qu, and Q. Mei. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *ACM SIGKDD*, 2015.

[124] L. Tang and H. Liu. Relational learning via latent social dimensions. In *ACM SIGKDD*, 2009.

[125] J. Teevan, S. T. Dumais, and E. Horvitz. Personalizing search via automated analysis of interests and activities. In *ACM SIGIR*, 2005.

[126] K.-C. Toh and S. Yun. An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. *Pacific Journal of Optimization*, 2010.

[127] J. Trajkova and S. Gauch. Improving ontology-based user profiles. In *Coupling approaches, coupling media and coupling languages for information retrieval*, 2004.

[128] K. E. Van De Sande, T. Gevers, and C. G. Snoek. Evaluating color descriptors for object and scene recognition. *TPAMI*, 2010.

[129] A. Van den Oord, S. Dieleman, and B. Schrauwen. Deep content-based music recommendation. In *NIPS*, 2013.

[130] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 2010.

[131] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *IEEE CVPR*, 2015.

[132] C. Wang and D. M. Blei. Collaborative topic modeling for recommending scientific articles. In *ACM SIGKDD*, 2011.

[133] C. Wang, Y. Song, A. El-Kishky, D. Roth, M. Zhang, and J. Han. Incorporating world knowledge to document clustering via heterogeneous information networks. In *ACM SIGKDD*, 2015.

[134] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu. Action recognition by dense trajectories. In *CVPR*, 2011.

[135] B. M. Whelan. *Color harmony 2: a guide to creative color combinations*. Rockport Publishers for Page One, 1994.

[136] D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *TEVC*, 1997.

[137] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust principal component analysis: Exact recovery of corrupted low-rank matrices by convex optimization. In *NIPS*, 2009.

[138] W. Wu, H. Li, H. Wang, and K. Q. Zhu. Probase: A probabilistic taxonomy for text understanding. In *ACM SIGMOD*, 2012.

[139] Z. Wu and M. Palmer. Verbs semantics and lexical selection. In *ACL*, 1994.

[140] S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha. Like like alike: joint friendship and interest propagation in social networks. In *ACM WWW*, 2011.

[141] X. Zabulis and S. C. Orphanoudakis. Image content analysis and description. In *State-of-the-Art in Content-Based Image and Video Retrieval*. Springer, 2001.

[142] H. Zhang, C. Y. Low, S. W. Smoliar, and J. H. Wu. Video parsing, retrieval and browsing: an integrated and content-based solution. In *ACM SIGMM*, 1995.

[143] H. Zhang, Z.-J. Zha, Y. Yang, S. Yan, Y. Gao, and T.-S. Chua. Attribute-augmented semantic hierarchy: Towards bridging semantic gap and intention gap in image retrieval. In *ACM SIGMM*, 2013.

[144] M.-L. Zhang and Z.-H. Zhou. Multilabel neural networks with applications to functional genomics and text categorization. *TKDE*, 2006.

[145] Z.-D. Zhao and M.-S. Shang. User-based collaborative-filtering recommendation algorithms on hadoop. In *WKDD*, 2010.

[146] V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang. Collaborative location and activity recommendations with gps history data. In *ACM WWW*, 2010.

[147] E. Zhong, N. Liu, Y. Shi, and S. Rajan. Building discriminative user profiles for large-scale content recommendation. In *ACM SIGKDD*, 2015.

[148] G. Zhu, S. Yan, and Y. Ma. Image tag refinement towards low-rank, content-tag prior and error sparsity. In *ACM SIGMM*, 2010.

[149] I. Zukerman and D. W. Albrecht. Predictive statistical models for user modeling. *UMUAI*, 2001.