ASPECT-BASED OPINION MINING IN ONLINE REVIEWS

by

Samaneh Abbasi Moghaddam

M.Sc., Sharif University of Technology, 2008

B.Sc., Ferdowsi University of Mashhad, 2005

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Doctor of Philosophy

in the
School of Computing Science
Faculty of Applied Sciences

SIMON FRASER UNIVERSITY
Spring 2013

APPROVAL

Name: Samaneh Abbasi Moghaddam

Degree: Doctor of Philosophy

Title of Thesis: Aspect-based Opinion Mining in Online Reviews

Examining Committee: Dr. Andrei Bulatov
Chair

Dr. Martin Ester, Professor, Computing Science
Simon Fraser University
Senior Supervisor

Dr. Fred Popowich, Professor, Computing Science
Simon Fraser University
Supervisor

Dr. Jian Pei, Professor, Computing Science
Simon Fraser University
SFU Examiner

Dr. Bing Liu, Professor, Computing Science,
University of Illinois at Chicago
External Examiner

Date Approved: March 22 2013

Abstract

Other people's opinions are important piece of information for making informed decisions. Today the Web has become an excellent source of consumer opinions. However, as the volume of opinionated text is growing rapidly, it is getting impossible for users to read all reviews to make a good decision. Reading different and possibly even contradictory opinions written by different reviewers even make them more confused. In the same way, monitoring consumer opinions is getting harder for the manufactures and providers. These needs have inspired a new line of research on mining consumer reviews, or opinion mining. Aspect-based opinion mining, is a relatively new sub-problem that attracted a great deal of attention in the last few years. Extracted aspects and estimated ratings clearly provides more detailed information for users to make decisions and for suppliers to monitor their consumers.

In this thesis, we address the problem of aspect-based opinion mining and seek novel methods to improve limitations and weaknesses of current techniques. We first propose a method, called Opinion Digger, that takes advantages of syntactic patterns to improve the accuracy of frequency-based techniques. We then move on to model-based approaches and propose an LDA-based model, called ILDA, to jointly extract aspects and estimate their ratings. In our next work, we compare ILDA with a series of increasingly sophisticated LDA models representing the essence of the major published methods in the literature. A comprehensive evaluation of these models indicates that while ILDA works best for items with large number of reviews, it performs poorly when the size of the training dataset is small, i.e., for cold start items. The cold start problem is critical as in real-life data sets around 90% of items are cold start. We address this problem in our last work and propose an LDA-based model, called FLDA. It models items and reviewers by a set of latent factors and learns them using reviews of an item category. Experimental results on real life data sets show that FLDA achieve significant gain for cold start items compared to the state-of-the-art models.

To my lovely husband, who supported me each step of the way!

“The more you know, the more you realize you know nothing!”

— SOCRATES

Acknowledgments

This research project would not have been accomplished without the support of many people. I wish to express my gratitude to my supervisor, Dr. Martin Ester who was abundantly helpful and offered invaluable assistance, support and guidance. Special thanks also to the members of the supervisory committee, Dr. Fred Popowich and Dr. Jian Pei without whose knowledge and assistance this study would not have been successful.

I would also like to express my gratitude to all my friends in the Department of Computing Science for their constant support and encouragement. Words are not enough to express my love and gratitude to my beloved husband for his endless love and great research collaborations through the duration of my studies. His constant support and encouragement under all odds has brought me where I stand today.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The emergence of user-generated content via social media has had an undeniable impact on the commercial environments. In fact, social media has shifted the content publishing from businesses towards the customers [18]. While there are several sources of user-generated content (e.g., discussion forums, Tweets, Blogs, News and reports, consumer feedback from emails, call centers, etc.), none of them is as focused as online reviews. That is why consumers mainly read and consult online reviews before purchasing products or using services. However, the growing volume of online opinions makes it harder and harder to make informed decisions. On the other hand, online reviews provide businesses with a rich source of consumer opinions for free. Marketing studies show that online reviews influence consumer shopping behavior significantly [89]. Many businesses are now tracking customer feedbacks through online sources. Amazon[1], Cnet[2], Rateitall[3], Epinions[4], TripAdvisor[5], Yelp[6], and Ciao UK[7] are examples of these Web resources containing consumer opinions.

These needs attracted researchers to a new field of study, called opinion mining, to analyze people's opinions, sentiments, evaluations, attitudes, and emotions from user generated content [92]. We start this dissertation with a discussion of demands for opinion mining in Chapter 2. We then define

[1] http://www.amazon.com/

[2] http://www.cnet.com/

[3] http://www.rateitall.com/

[4] http://www.epinions.com/

[5] http://www.tripadvisor.com

[6] http://www.yelp.com/

[7] http://www.ciao.co.uk/

the terminologies used in this area and discuss general opinion mining tasks. Based on the problem classifications presented in the literature [128, 90, 91], we introduce an extended categorization for opinion mining tasks by adding the emerged research topics in the last few years. This categorization is organized at three levels of granularity: document-level, sentence-level, and phrase-level tasks. In document-level tasks the input document should be classified in a few predefined categories, e.g., subjectivity classification. The tasks at the sentence level go to the sentences, e.g., opinion mining in comparative sentences. Finally, phrase-level opinion mining performs fine-grained analysis and directly looks at the opinion not at language constructs, e.g., aspect-based opinion mining.

Aspect-based opinion mining aims to extract major aspects of an item and to predict the rating of each aspect from the item reviews. Aspects are attributes or components of items (e.g., 'LCD', 'battery life', etc. for a digital camera) and ratings are the intended interpretation of user satisfaction in terms of numerical values. In this thesis, we focus on the problem of aspect-based opinion mining because of its key role in the area of opinion mining. The importance of this problem is not only due to easing the process of decision making for customers by providing a decomposed view of rated aspects, but also due to the ability of utilizing the extracted rated aspects in other opinion mining systems, e.g., opinion summarization, opinion question answering, etc. In Chapter 3, we first define the problem of aspect-based opinion mining and discuss its main challenges. We also review the evaluation metrics that have been used in the literature and survey the publicly available data sets. Then, we present a comprehensive review of the state-of-the-art methods according to the following categorization: frequency- and relation-based approaches and model-based approaches.

In Chapter 4, we introduce Opinion Digger to mine and summarize opinions from customer reviews [109]. It takes advantage of both frequency- and relation-based approaches to identify aspects and estimate their rating. Opinion Digger first mines a set of opinion patterns (representing the relationship between aspects and sentiments) from the review texts and then uses it for filtering frequent noun phrases. It also uses a novel technique to group synonymous aspects. In addition, in contrast to the previous methods that mostly determine whether an opinion is positive or negative, Opinion Digger determines the strength of positiveness and negativeness of each aspect by assigning a numerical rating in the range from 1 to 5. We conduct experiments on a real life dataset from Epinions.com, demonstrating the improved effectiveness of Opinion Digger in terms of accuracy of aspect extraction. Evaluation of rating estimation and candidate grouping also demonstrate the high accuracy of the proposed techniques.

While frequency- and relation-based methods are quite effective, they require the manual tuning of various parameters which makes them hard to port to another dataset. Model-based techniques overcome this limitation by automatically learning the model parameters from the data. So, recently researchers have explored unsupervised learning techniques, especially topic modeling, for aspect-based opinion mining. Different extensions of the basic topic models have been proposed to address the problem of aspect-based opinion mining. However, all of these models perform aspect identification and rating prediction in separate steps leading to the accumulation of errors. In Chapter 5, we introduce Interdependent Latent Dirichlet Allocation (ILDA) to jointly identify aspects and predict their ratings from online reviews. We argue that ILDA is most natural for our problem since the underlying probabilistic assumptions (interdependency between aspects and ratings) are appropriate for the problem domain. We conduct experiments on a real life dataset from Epinions.com, demonstrating the improved effectiveness of the ILDA model in terms of the likelihood of a held-out test set, and the accuracy of aspects and aspect ratings. We also evaluate the performance of our model on real-life data sets from Amazon.com and TripAdvisor.com.

In Chapter 6, we compare ILDA with a series of increasingly sophisticated LDA models representing the essence of the major published methods. This comprehensive comparison allows us to tease apart the impact of various design decisions and derive some guidelines for designing future models. We evaluate the performance of these models in terms of the likelihood of a held-out test set on a very large real-life data set from Epinions.com. As a novel technical contribution, we present a method for preprocessing reviews based on grammatical relations provided by a dependency parser. This method promises to generate opinion phrases more accurately than current methods that consider only syntactic properties such as the proximity of words. We also measure the impact of the training set size and perform experiments for different subsets of items with different numbers of reviews. The results indicate that while ILDA works best for items with large number of reviews, it performs poorly when the size of the training dataset is small, i.e., for cold start items.

The cold start problem is critical as in real-life data sets, such as those from Epinions.com and Amazon.com, where more than 90% of items are cold start. We address this problem in Chapter 7 and propose a probabilistic graphical model based on LDA, called FLDA. FLDA models items and reviewers by a set of latent factors and learns them using reviews of an item category. Experimental results on real life data sets from Epinions.com, Amazon.com, and TripAdvisor.com show that our proposed model achieve significant quality gain for cold start items compared to the state-of-the-art models. Finally, we conclude the thesis in Chapter 8 with a summary and discussion of promising directions for future research.

Chapter 2

Introduction to Opinion Mining

Most of the current text processing methods (e.g. search engines, question answering systems, text mining tools, etc.) work with factual information. However, the Web contains huge volumes of opinionated text. Internet users express personal experiences and opinions on almost anything at review websites, discussion groups, forums, blogs, etc. This valuable information is publicly available for Internet users.

However, the large collection of opinions on the Web makes it very difficult to get useful information easily. Reading all reviews to make an informed decision is a time-consuming job. Reading different and possibly even contradictory opinions written by different reviewers may make businesses/customers more confused. These needs have inspired a new line of research on mining customer opinions, which is called *opinion mining*.

Opinion mining is the field of study that analyzes people's opinions, sentiments, evaluations, attitudes, and emotions from written language [92]. It has attracted a lot of researchers from different areas of research including natural language processing, data mining, machine learning, linguistics, and even social science.

In this chapter, we first discuss demands for opinion mining and then define the terminologies used in this area. In section 2.3, we discuss general opinion mining tasks and present a brief review of the existing and related works on each task. We categorize opinion mining tasks into nine classes (under three general categories) by extending the most current categorization schemas presented in the literature [92].

2.1 Demands for Opinion Mining

With the growth of social media services such as review sites, forum discussions, blogs, micro-blogs, and online social networks, the interest in opinion mining has increased significantly. Today online opinions have turned into a kind of virtual currency for businesses looking to market their products, identify new opportunities and manage their reputations [1]. Many businesses are now employing opinion mining techniques to track customer feedbacks to action appropriately.

Recently one of Canada's largest Internet marketing companies reported some statistics on the review revolution[2]:

- Traffic to the top 10 review sites grew on average 158% last year.
- 92% of online consumers have more confidence in info found online than they do in anything from a salesclerk or other source.
- 70% consult reviews or ratings before purchasing.
- 97% who made a purchase based on an online review, found the review to be accurate.
- 7 in 10 who read reviews share them with friends, family and colleagues thus amplifying their impact.
- 34% have turned to social media to share their feelings about a company. 26% to express dissatisfaction, 23% to share companies or products they like.

Pang et al. [128] also report the results of two surveys on American adults which show strong demand for opinions. The authors summarize the surveys' findings as follows:

- 81% of Internet users have done online research on a product at least once.
- 20% do online research every day.
- Between 73% and 87% report that reviews had a significant influence on their purchase.

[1] Wikipedia. Sentiment analysis. `http://en.wikipedia.org/wiki/Sentiment_analysis`. [Online; accessed 2-January-2013].

[2] `http://www.searchenginepeople.com/blog/12-statistics-on-consumer-reviews.html#ixzz2DkFWhnBg`

- Respondents are willing to pay from 20% to 99% more for a 5-star-rated item than a 4-star-rated item.
- 32% provided a rating on a product, service, or person in an online rating system at least once.
- 30% posted an online comment or review regarding a product or service.

The authors not only disclose users "need" for online opinions, but also report that 58% of people indicate that online information was missing, impossible to find, confusing, and/or overwhelming. These results reveal the need for better opinion retrieval and opinion mining systems.

Opinion mining is not only very useful for consumers to know the opinions of other users before they use a service or purchase a product, but also crucial for businesses to understand consumer opinions on their products and services [90]. While product specifications are obviously relevant, finding the main reason for low sales requires focusing more on people's personal views of such characteristics [128]. Opinion mining is an excellent tool for handling many business tasks related to sales management, reputation management, and public relations. Moreover, companies may be able to perform trend prediction in sale by tracking consumer viewpoints.

2.2 Opinion Mining Terminologies

In this section we define the basic terminologies currently used in the area of opinion mining.

Fact: A fact is something that has really occurred or is actually the case [3].

Opinion: An opinion is a belief about matters commonly considered to be subjective, and is the result of emotion or interpretation of facts [4].

Subjective/Opinionated Text: A text is subjective or opinionated if it expresses personal feelings or beliefs, e.g. opinions.

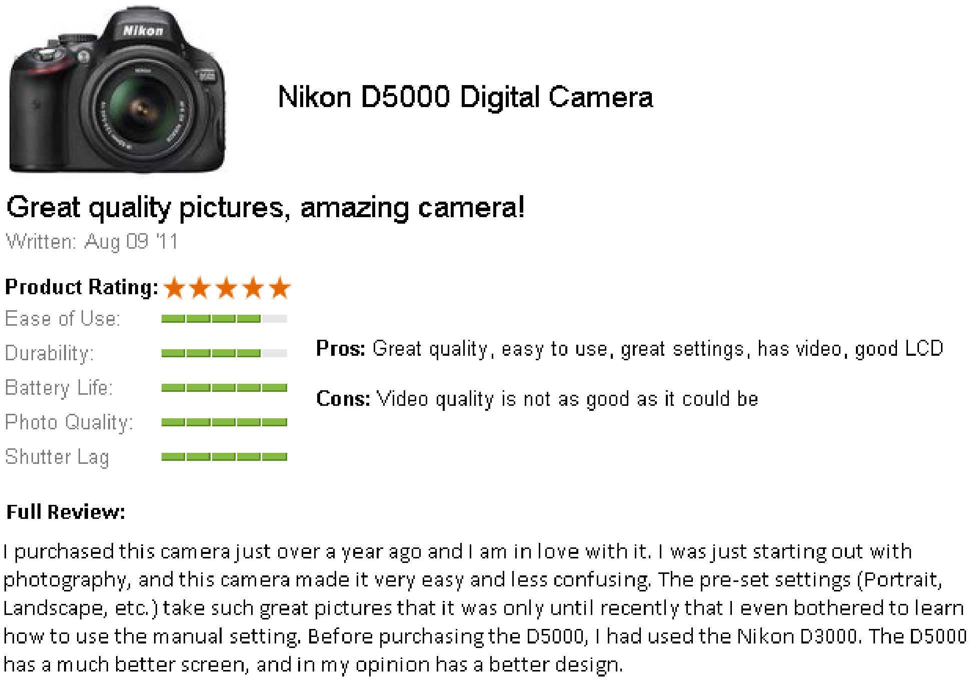
Objective Text: An objective text expresses some factual information about the world.

Item: An item is a concrete or abstract object such as product, service, person, event, organization [92]. An item can be represented as a hierarchy of components, sub-components, etc.

Review: A review is a subjective text containing a sequence of words describing opinions of reviewer regarding a specific item. Review text may contain complete sentences, short comments, or both (Figure 2.1).

[3] Wikipedia. Fact. `http://en.wikipedia.org/wiki/Fact`. [Online; accessed 2- January-2013].

[4] Wikipedia. Opinion. `http://en.wikipedia.org/wiki/Opinion`. [Online; accessed 2-January-2013].



Nikon D5000 Digital Camera

Great quality pictures, amazing camera!

Written: Aug 09 '11

Product Rating: ★★★★★

Ease of Use:	🟩🟩🟩🟩⬜
Durability:	🟩🟩🟩🟩⬜
Battery Life:	🟩🟩🟩🟩🟩
Photo Quality:	🟩🟩🟩🟩🟩
Shutter Lag	🟩🟩🟩🟩🟩

Pros: Great quality, easy to use, great settings, has video, good LCD

Cons: Video quality is not as good as it could be

Full Review:

I purchased this camera just over a year ago and I am in love with it. I was just starting out with photography, and this camera made it very easy and less confusing. The pre-set settings (Portrait, Landscape, etc.) take such great pictures that it was only until recently that I even bothered to learn how to use the manual setting. Before purchasing the D5000, I had used the Nikon D3000. The D5000 has a much better screen, and in my opinion has a better design.

Figure 2.1: A sample review from Epinions.com

- Short comments or Pros/Cons: The reviewer can describe Pros and Cons of the item separately. Short comments contain sentence segments that are usually separated by comas, e.g., “great movie quality, poor LCD display, affordable price”.

- Full text review or free format: The reviewer can write freely, i.e., no separation of Pros and Cons. Full text reviews contain complete sentences and tend to be long. The sentences are usually complex and have a large amount of irrelevant information, e.g., “When I bought it I was not quite sure it is the best choice, but now I am pretty sure it is the best camcorder with great movie quality and long lasting battery life”.

Aspect: An aspect (also called feature) is an attribute or component of the item that has been commented on in a review. If an aspect appears in a review, it is called explicit aspect; otherwise it is called implicit [90]. Current works mainly focus on extracting explicit aspects and only a few simple methods are proposed for identifying implicit aspects.

- Explicit Aspects: Aspects that are explicitly mentioned as nouns or noun phrases in a sentence, e.g., ‘picture quality’ in the sentence “The picture quality of this phone is great”.

- Implicit Aspects: Aspects that are not explicitly mentioned in a sentence but are implied, e.g., ‘price’ in the sentence “This car is so expensive.”, or ‘size’ in the sentence “This phone will not easily fit in a pocket”.

Known Aspects: Known aspects are predefined aspects provided by the reviewing website for which users explicitly express ratings, e.g., ease of use, durability, etc. in Figure 2.1.

Sentiment: Sentiment is a linguistic term which refers to the direction in which a concept or opinion is interpreted [90]. We use sentiment in a more specific sense as an opinion about an aspect. For example, ‘great’ is a sentiment for the aspect ‘picture quality’ in the sentence “It has great picture quality”.

Opinion Phrase: An opinion phrase $< h, m >$ is a pair of head term h and modifier m [101]. Usually the head term is a candidate aspect, and the modifier is a sentiment that expresses some opinion towards this aspect, e.g. $<$ *LCD*, *blurry* $>$, $<$ *screen*, *inaccurate* $>$, etc.

Opinion Orientation: A sentiment can be classified in n-level orientation scale. Sentiment orientation is an intended interpretation of the user satisfaction in terms of numerical values.

- Polarity: Polarity is a two-level orientation scale. In this scale a sentiment is either positive or negative.

- Rating: Most of the reviewing websites use five-level orientations, presented by stars in the range from 1 to 5 which is called rating.

Overall Rating: Most of the review websites ask reviewers to express an overall rating (as stars) for the reviewed item indicating the overall quality of the reviewed item (e.g., Product Rating in Figure 2.1).

Rating Guideline: Some of the reviewing websites provide some guidelines for users in assigning overall ratings. For example, Epinions provides a rating guideline stating that “rating 5 means excellent, rating 4 means good, rating 3 means average, rating 2 means poor, and rating 1 means terrible”.

Part-of-Speech (POS) Tag: The part-of-speech of a word is a linguistic category that is defined by its syntactic or morphological behavior. Common POS categories in English grammar are: noun, verb, adjective, adverb, pronoun, preposition, conjunction, and interjection. POS tagging is the task of labeling (or tagging) each word in a sentence with its appropriate part of speech. Most of the opinion mining works use the standard Penn Treebank POS Tags[5]. Table 2.1 shows some of the

[5] http://www.cis.upenn.edu/~treebank/home.html

Table 2.1: Part of the Penn Treebank part-of-speech tags

Tag	Description	Tag	Description
DT	Determiner	CC	Coordinating Conjunction
JJ	Adjective	RB	Adverb
NN	Noun, singular or mass	VB	Verb, base form

common POS tags.

2.3 General Opinion Mining Tasks

In this section we present a review of the existing and related works on opinion mining proposed in the literature. For a comprehensive review, we categorize opinion mining tasks and methods into three general classes, but before we discuss our categorization, we present current categorization schemas encountered in the literature.

Pang et al. [128] group the major problems of opinion mining into three classes: sentiment polarity identification, subjectivity detection, and joint topic-sentiment analysis. Liu [90] also defines three mining tasks for opinionated text in his book. He further extends this categorization in his handbook [91] as follows: sentiment and subjectivity classification, aspect-based opinion mining, sentiment analysis of comparative sentences, opinion search and retrieval, and opinion spam detection. Finally, in his recent book [92] he defines three general categorizations for opinion mining tasks: document-level, sentence-level, and phrase-level.

Inspired by the above problem classifications, we introduce an extended categorization for opinion mining tasks by adding the emerged research topics in the last few years:

- Document-level opinion mining
  - Subjectivity classification
  - Sentiment classification
  - Opinion quality and helpfulness estimation
  - Opinion spam detection
- Sentence-level opinion mining
  - Opinion search and retrieval

  - Opinion question answering
  - Opinion summarization
  - Opinion mining in comparative sentences

- Phrase-level opinion mining
  - Aspect-based opinion mining

It should be noted that the tasks at each level might also be applied at other levels. For example, subjectivity and sentiment classification can also be performed at sentence level, opinion search and retrieval can be performed at document-level, and opinion summarization may be applied at phrase-level. We categorize each task into one of these levels based on majority of works proposed for that task. In the next section we will review the most important approaches and techniques proposed for each task.

2.3.1 Document-Level Opinion Mining

Document-level tasks are mainly formulated as classification problems where the input document should be classified into a few predefined categories. In subjectivity classification, a document is classified as subjective or objective. In sentiment classification, a subjective document is classified as positive, negative, or neutral. Opinion helpfulness prediction classifies an opinion as being helpful or not helpful (sometimes more classes are defined). Finally, opinion spam detection classifies opinions as spam and not spam.

Subjectivity Classification

Subjectivity classification aims to determine whether a given text expresses an opinion or not. In other words, whether a text is factual (objective) or opinionated (subjective). This problems is usually considered as a classification problem. Most of the existing techniques are based on supervised learning [141, 177, 176], although there are some unsupervised methods.

One of the early works in this area presented by Wiebe et al. [175] performs subjectivity classification using the naïve Bayesian classifier. Subsequent research uses other learning algorithms [141, 177, 176] for identifying opinionated text. Later research has been mainly focused on developing methods for multilingual subjectivity analysis [69, 8]. One of the difficulties in subjectivity classification is the manual effort involved in labeling training examples as subjective or objective.

As a result, some research have also focused on developing methods to label training data automatically [142, 143].

Sentiment Classification

This is the area that has been researched the most in academia. Sentiment classification assumes that the given document is opinionated and aims to find the general opinion of the author in the text [162]. For example, given a product review, it determines whether the review is positive or negative. Sentiment classification, in contrast to subjectivity analysis, does not usually need manual effort for annotating training data. Training data used in sentiment classification are mostly online product reviews that have already been labeled by reviewers with the assigned overall ratings (usually in the range form 1 to 5). Typically a review with 4-5 stars are considered positive, and a review with 1-2 stars are considered negative [91].

Current works mainly apply supervised learning methods to sentiment classification [126]. As one of the early works, Pang et al. [130] apply three machine-learning methods (naïve Bayes, Maximum Entropy classification, and Support Vector Machine (SVM)) to classify movie reviews as positive or negative. They show that the standard machine learning techniques outperform human-produced baselines. Subsequent works use many more kinds of classification features (e.g. terms and their frequency, part of speech tags, opinion words and phrases, etc.) and techniques in learning [105, 168, 127].

There are also some unsupervised methods for classifying reviews [166, 26, 184, 27]. These works introduce different score functions for classifying a review as positive or negative (thumbs up or down). These algorithms mainly compute semantic orientation of document terms (sometimes only opinion terms, e.g. adjectives and adverbs) using the defined score functions. Then documents are classified by averaging the orientation of their phrases. Recently researchers also show interest in sentiment classification at finer grained level (e.g. sentences, and phrases) [103, 161, 140, 160, 44] and building lexical resources for opinion mining [30, 48, 4, 135, 98, 57].

Opinion Quality and Helpfulness Estimation

The problem of automatically evaluating the helpfulness of online reviews has also attracted increasing attention. Most previous works attempt to predict the helpfulness of reviews by using a set of observed features, i.e. textual and/or social features and learning a function of these features for predicting review helpfulness. Textual features include features that are based on text statistics, such as length of the review, the average length of a sentence, percentage of nouns or adjectives, etc. Social features, on the other hand, are related to the author of the review and are extracted from his social context, such as the number of past reviews by the author, in-degree and out-degree of the author in the social network, past average rating for the author, etc.

Current works have formulated the problem of evaluating review helpfulness as a classification or regression problem using these observed features. The ground truth data used for both training and testing are usually the user helpfulness feedback (vote) given to each review in the reviewing Websites. The authors of [124, 94, 151, 174] propose classification-based approaches learning from annotated ground-truth. Each work define and use a set of features in the process of estimating review helpfulness. Instead of classifying reviews as helpful or unhelpful, some recent works consider estimating a helpfulness rating for each review using a regression model [71, 185, 40, 193, 100, 125] or a factor model [116, 115]. There are also some analytical studies on the helpfulness of reviews [94, 24, 52]. In [94] the “rich-get-richer” effect is identified, where reviews accumulate helpfulness votes more quickly depending on the number of votes they already have. This analysis strengthens the motivation for automatically determining the helpfulness of reviews in order to avoid such biases.

Opinion Spam Detection

Web spam is quite familiar to most people. In the context of opinion, we have a similar spam problem [62]. The growth of user-generated contents motivates more people to find and read opinions on the Web. Positive opinions can result in significant financial gain or fame for the manufacturers or vendors, and negative opinions, on the other hand, can have the reverse impact on them. This impact also gives good incentives for writing spam reviews (opinion spam) [91]. Spam opinions try to deliberately mislead readers by giving undeserving positive or negative opinions to some target items in order to promote the item and/or by giving unjust negative opinions to some other items to damage their reputation [61].

Spam detection can be formulated as a classification problem with two classes: spam and non-spam. The main task is to find the set of effective data features for building the model. Jindal et al. [61, 62] identify some textual and social features for learning a regression model to estimate the probability of each review being a spam. In [63, 87] different methods for identifying spam reviews using behavior of reviewers are proposed. There are also some works on identifying opinion leader [85], and detecting group review spam [119].

2.3.2 Sentence-Level Opinion Mining

The tasks at this level go to the sentences. In opinion search and retrieval and opinion question answering, sentences are usually retrieved and ranked based on some criteria. Opinion summarization aims to select a set of sentences (or phrases) which summarize the opinion more accurately. Finally, opinion mining in comparative sentences includes identifying comparative sentences and extracting information from them.

Opinion Search and Retrieval

As traditional Web search is very important for Internet users, opinion search will be also of great use. Searching the user-generated content on the Web enables users to find opinions on any subject matters. Opinion search queries are mainly issued to find public opinion on a particular item or an aspect of the item. For example, to find public opinion on a digital camera or the picture quality of a camera, a user may issue the query “camera X picture quality”.

Similar to traditional Web search, opinion search has two main tasks: retrieving relevant text (document, passage, sentence) to the user query, and ranking the retrieved text. However, there is a major difference in retrieving phase of opinion search. The retrieved text in an opinion search method needs to be not only relevant to the user query, but also opinionated. Some of the methods first extract relevant documents and then filter out objective ones [51, 192], while others first identify opinionated documents and then find relevant text to the query among them [37]. The authors of [191, 33, 39, 53] also present probabilistic models that unifies topic relevancy and opinionatedness for retrieving documents.

Regarding the ranking task, traditional Web search engines usually rank Webpages based on authority and relevance score. The assumption is that the top ranked pages contain sufficient information to satisfy the user’s information need. However, this assumption is not true in the domain of opinions. The top ranked documents only represent the opinions of few persons not the public. The ranked results of an opinion search engine needs to reflect the natural distribution of positive and negative sentiments of the whole population [90]. Current ranking methods use different criteria to reflect the public opinion. The method proposed in [80] uses the behavioral model of consumers using economic approach for ranking products. In other works, review quality [51, 107], text statistics (e.g. number of terms, similarity score) [129, 192], user feedback [120] and recency of reviews [31] are considered as measures of ranking.

Opinion Summarization

Text summarization involves reducing a text document or a larger corpus of multiple documents into a short set of words or sentences that conveys the main meaning of the text. Opinion summarization, is expected to allow all possible reviews to be efficiently utilized [122]. Two particular types of opinion summarization that often addressed in the literature are keywords extraction and review summarization. The goal of keywords extraction is to select individual words or phrases to provide a summary of reviews as a structured table [48, 101, 94]. Since the main task of these methods are identifying aspects and sentiments, we discuss these approaches further in Section 2.3.3 where we discuss aspect-based opinion mining techniques.

The goal of review summarization, on the other hand, is selecting whole sentences to create a short paragraph summary. Given multiple reviews, a review summarization method outputs text consisting of ordered sentences. Existing approaches focus on selecting sentences so that the summary includes important information of the reviews [95, 106, 165]. However, most of them ignore the coherence of the summary, which negatively affects readers' comprehension. Recently the authors of [122, 123] propose ranking algorithms to order the extracted sentences appropriately to keep the coherence between sentences.

Opinion Question Answering

Another interesting task in opinion mining is opinion Question Answering (opinion QA). Opinion QA methods try to answer opinion-based questions using reviewers' opinions about target items [95, 110]. They are often more complicated than traditional QA problems which resolve fact-based questions, e.g. "What is the longest river in the world?". In contrast to fact-based QA systems that seek related factual information to the given question, opinion QA methods aim to find authors' sentimental opinion on a specific target, e.g. "Do people recommend digital camera X?"

As discussed in [156], answers of opinion questions are usually longer and more likely to be partial. So, in contrast to fact-based questions, opinion questions usually do not have unique answers. The authors of [156, 7, 6] compare and contrast the properties of fact and opinion questions and answers. They identify and explore the challenges raised by opinion question answering, as opposed to the factual QA. Based on the different characteristics, they conclude that traditional QA approaches are not as effective for opinion questions as they have been for fact-based questions.

Most of the current methods [76, 181, 153] develop a classifier for discriminating between subjective (opinion-based) and objective (fact-based) documents and then apply some techniques for detecting opinion sentences in documents, e.g. similarity approach. There are also some works on collaborative question answering [84, 158, 79] where questions and answers are both given as input and the task is to find an appropriate list of answers for a given question, e.g. Yahoo! Answers.

Opinion Mining in Comparative Sentences

In general, a comparative sentence is a sentence that expresses a relation based on similarities or differences of more than one item [59]. The comparison in a comparative sentence is usually expressed using comparative (e.g. ‘smaller’, ‘better’) or superlative (e.g. ‘the smallest’, ‘the best’) forms of an adjective or adverb. While little research has been done in this area of research, we can identify two main tasks in comparison mining: identifying comparative sentences in the given opinionated text, and extracting comparative opinion from the identified sentences.

Identifying comparative sentences is usually treated as a classification problem and a machine learning algorithm is applied to solve the problem [78]. The second task involves extracting items and their aspects that are being compared, and the comparative keywords. For extracting items and their aspects being compared, different information extraction methods can be applied, e.g. Conditional Random Field (CRF).

One of the early works in this area is presented by Jindal et al. [59, 60]. They manually collect a set of comparative and superlative adjectives and adverbs and then extract a set of POS-patterns using these keywords to identify comparative sentences. A slightly different method is proposed in [68]. In this work identifying comparative sentences is framed as an optimization problem. The optimization framework is based on two basic similarity measures defined on pair of sentences. There are also some works considering a sub-problem of this area [16, 169]. The authors of [36] study the problem of identifying the product that has more of a certain aspect in a comparative sentence, while that of [38] focus on determining the product that is preferred by the reviewers.

2.3.3 Phrase-Level Opinion Mining

Document-level and sentence-level analyses do not discover what exactly people liked and did not like. Phrase-level opinion mining performs finer-grained analysis and directly looks at the opinion [92]. The goal of this level of analysis is to discover sentiments on aspects of items.

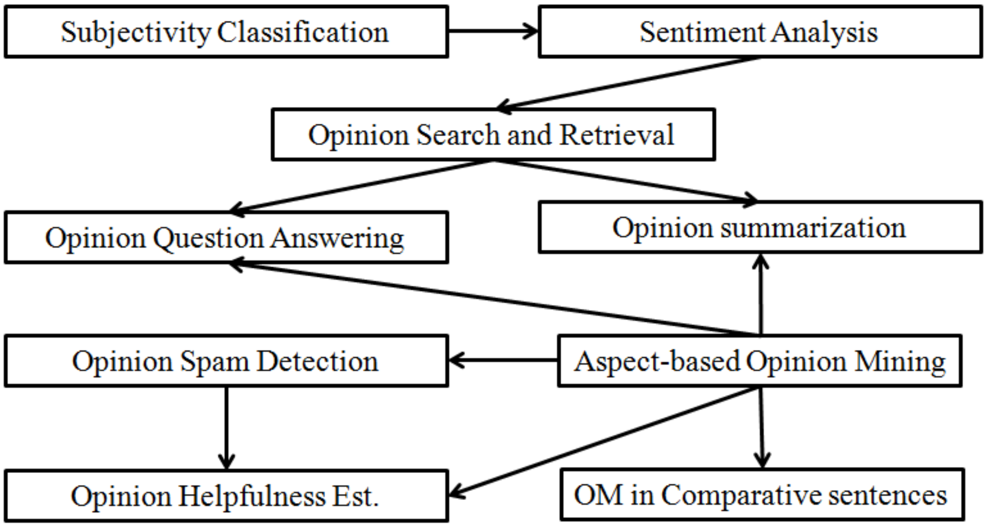

Subjectivity Classification
Sentiment Analysis
Opinion Search and Retrieval
Opinion Question Answering
Opinion summarization
Opinion Spam Detection
Aspect-based Opinion Mining
Opinion Helpfulness Est.
OM in Comparative sentences

Figure 2.2: Relationships among opinion mining tasks

Aspect-Based Opinion Mining

Recently, aspect-based opinion mining has attracted a lot of attention. Several methods have been proposed to extract aspects from reviews. Some of these works used full text reviews, while others took advantages of more structured short comments in this process. Multiple algorithms have also been presented for identifying the rating of aspects. Extracted aspects and estimated ratings clearly provides more detailed information for users to make decisions and for suppliers to monitor their consumers.

Since, aspect-based opinion mining is the main focus of this thesis, we omit the brief literature review in this section, and instead dedicate a whole chapter to this topic. In the next chapter, we first define the problem of aspect-based opinion mining formally, and then extensively review state-of-the-art approaches. We focus on this problem because of its key role in the area of opinion mining. The importance of this problem is not only due to easing the process of decision making for customers by providing a decomposed view of rated aspects, but also due to the ability of utilizing the extracted rated aspects in other opinion mining systems (Figure 2.2), e.g.:

- Opinion summarization systems can use rated aspects to find sentences which summarize the review more accurately.
- Opinion question answering systems benefit from using aspect ratings when answering comparative questions.

- Estimating review helpfulness can also be improved by considering the percentage of covered aspects in a reviews and also the conformity of aspect ratings with the crowd.

The results of aspect-based opinion mining can also be used in other computer systems, such as recommendation systems (to provide explanations for recommendation), advertising system (to place an ad of a product with similar rated aspects), and many business tasks related to sale management, reputation management, and public relations.

Chapter 3

Aspect-based Opinion Mining

Mining opinions at the document-level or sentence-level is useful in many cases. However, these levels of information are not sufficient for the process of decision-making (e.g. whether to purchase the product). For example, a positive review on a particular item does not mean that the reviewer likes every aspect of the item. Likewise, a negative review does not mean that the reviewer dislikes everything. In a typical review, the reviewer usually writes both positive and negative aspects of the reviewed item, although his general opinion on the item may be positive or negative. In fact, document-level and sentence-level opinions cannot provide detailed information for decision-making. To obtain such information, we need to go to a finer level of granularity.

In the past decade a large number of methods have been proposed for the problem of aspect-based opinion mining. The earliest works are frequency-based approaches where simple filters are applied on high frequency noun phrases to extract aspects. While these methods are quite effective, they miss low frequency aspects. To overcome this weakness, relation-based techniques are proposed. These methods use Natural Language Processing (NLP) techniques to find some relationships between aspects and related sentiments (adjectives describing aspects' quality). While they overcome the weakness of the frequency-based methods, they produce many non-aspects matching with the NLP relations. Finally, researchers took advantages of both approaches and proposed hybrid techniques which use NLP relations for filtering high frequency candidates. The accuracy of hybrid methods is much higher than the previous methods. However, similar to the previous approaches, hybrid methods need manual tuning of various parameters that make them hard to port to another dataset.

To avoid the need for manually tuning parameters, researchers explored supervised learning techniques that automatically learn model parameters from the data. In fact, they assume that aspect-based opinion mining can be seen as a special case of the general information extraction problem and supervised learning techniques can be applied to reviews to identify aspects and their ratings. While this approach overcomes the weaknesses of the previous approaches, it still needs manually labeled data for training the models. To address this weakness, researchers have recently explored unsupervised learning techniques, especially topic modeling, assuming that topics from topic models can represent aspects and sentiments. To separate these two, different extensions of basic topic models have been proposed.

In this chapter, we first define the problem of aspect-based opinion mining more formally. Then we present a comprehensive review of the state-of-the-art methods. We also discuss the main challenges of this problem and review the evaluation metrics that have been used in the literature. In addition, we survey the publicly available data sets and discuss their properties.

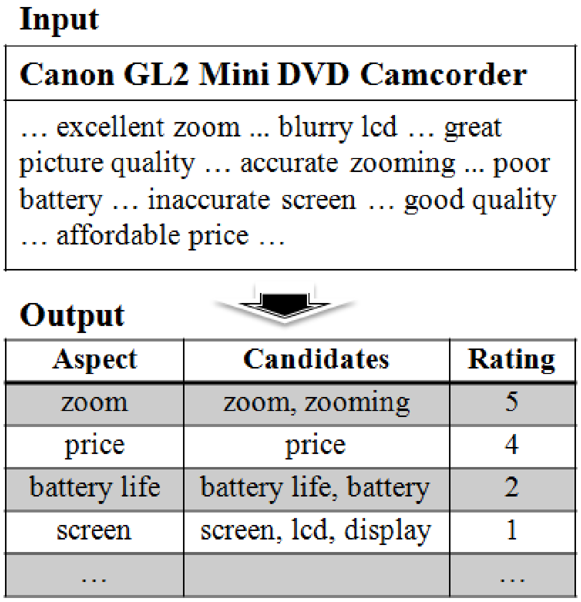
3.1 Problem Statement

Aspect-based opinion mining addresses the needs for detailed information. In the last decade, several methods have been proposed to extract aspects from reviews, e.g., [48, 93, 132, 163, 179, 101, 170]. Some of these works used full text reviews which usually have a large amount of irrelevant information, while others took advantages of short comments. Multiple algorithms have also been presented for identifying the rating of aspects. Extracted aspects and estimated ratings clearly provide more detailed information for users to make decisions and for suppliers to monitor their consumers.

Problem Definition: Given a set of reviews about item P, the task is to identify the k major aspects of P and to predict the rating of each aspect [112]. The problem definition is illustrated in Figure 3.1. In general, aspect-based opinion mining consists of two main tasks:

- Aspect identification: The goal of this task is to extract aspects of the reviewed item and to group synonyms of aspects, as different people may use different words or phrases to refer to the same aspect, e.g., display, screen, LCD.

- Rating (polarity) prediction: This task aims at determining whether the opinion on the aspect is positive/negative or estimating the rating of the opinion in a numerical range (usually in the range from 1 to 5).



Input

Canon GL2 Mini DVD Camcorder

… excellent zoom ... blurry lcd … great picture quality … accurate zooming ... poor battery … inaccurate screen … good quality … affordable price …

Output

Aspect	Candidates	Rating
zoom	zoom, zooming	5
price	price	4
battery life	battery life, battery	2
screen	screen, lcd, display	1
…		…

Figure 3.1: Aspect-based opinion mining problem definition

A Motivating Example

A sample of extracted aspects and their estimated ratings for two camcorders is shown in Table 3.1. As shown, although these two camcorders have the same overall ratings, Camcorder X has better zoom and battery life, while Camcorder Y has better screen and sound. This decomposed view of rated aspects clearly provides more detailed information than the overall rating and helps users to make better decisions.

Table 3.1: Decomposed view of rated aspects for two camcorders

Aspects	Camcorder X	Camcorder Y
zoom	4	2
sound	1	3
screen	2	4
price	3	3
battery life	4	2
Overall Rating	3	3

Extracted aspects and estimated ratings can also be used as input for various computer systems:

- Summarization systems, to find sentences which summarize the review more accurately.
- Recommendation systems, to provide explanations for recommendation.
- Question answering systems, to answer opinion-based questions by comparing aspects and ratings of different products.
- Opinion helpfulness estimation, to estimate the helpfulness based on the percentage of covered aspects and the conformity of their ratings with the crowd.

A Real-life Application

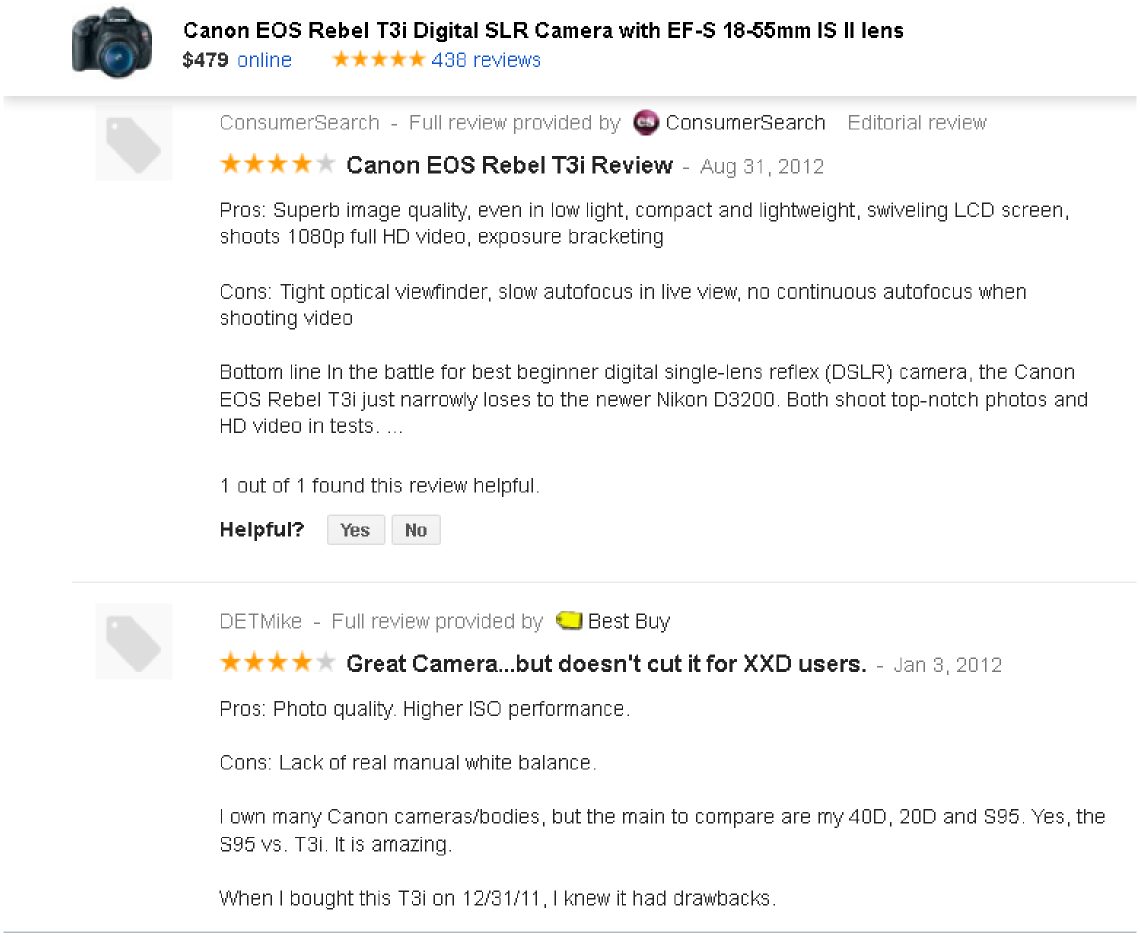
“Google Shopping”[1], formerly “Google Product Search” is an internet marketplace launched by Google Inc. Users can type product queries to return lists of vendors selling a particular product, as well as pricing information, product overall rating and product reviews. Product reviews in Google Shopping are from third party sites. For example reviews of a digital camera are gathered from ConsumerSearch.com, BestBuy.com, Epinions.com, NewEgg.com, etc. (Figure 3.2).

In addition to listing the review texts, Google Shopping applied an aspect-based opinion mining technique to extracts product aspects from reviews. It also presents the percentages of positive and negative sentences for each extracted aspects to help users in decision making. In Figure 3.3 the percentage of positive and negative sentences for each aspect is shown by green and red colors, respectively. For each aspect, one representative sentence is shown. By clicking on the aspect a list of sentences containing that aspects in different reviews are also listed.

3.2 State-of-the-art Aspect-based Opinion Mining Techniques

As discussed in the previous section, there are two main tasks in the problem of aspect-based opinion mining: aspect extraction, and rating prediction. Bing Liu in his recently published book [92] classified aspect extraction techniques into four categories: finding frequent nouns and noun phrases, using opinion and target relations, using supervised learning, and using topic models. We extend this categorization by adding a category for hybrid approaches which make use of opinion relations for filtering frequent noun phrases and introduce the following categorization for aspect extraction techniques [114]:

[1] `http://www.google.com/shopping`



Canon EOS Rebel T3i Digital SLR Camera with EF-S 18-55mm IS II lens
$479 online ★★★★★ 438 reviews

ConsumerSearch - Full review provided by ConsumerSearch Editorial review

★★★★☆ **Canon EOS Rebel T3i Review** - Aug 31, 2012

Pros: Superb image quality, even in low light, compact and lightweight, swiveling LCD screen, shoots 1080p full HD video, exposure bracketing

Cons: Tight optical viewfinder, slow autofocus in live view, no continuous autofocus when shooting video

Bottom line In the battle for best beginner digital single-lens reflex (DSLR) camera, the Canon EOS Rebel T3i just narrowly loses to the newer Nikon D3200. Both shoot top-notch photos and HD video in tests. ...

1 out of 1 found this review helpful.

DETMike - Full review provided by Best Buy

★★★★☆ **Great Camera...but doesn't cut it for XXD users.** - Jan 3, 2012

Pros: Photo quality. Higher ISO performance.

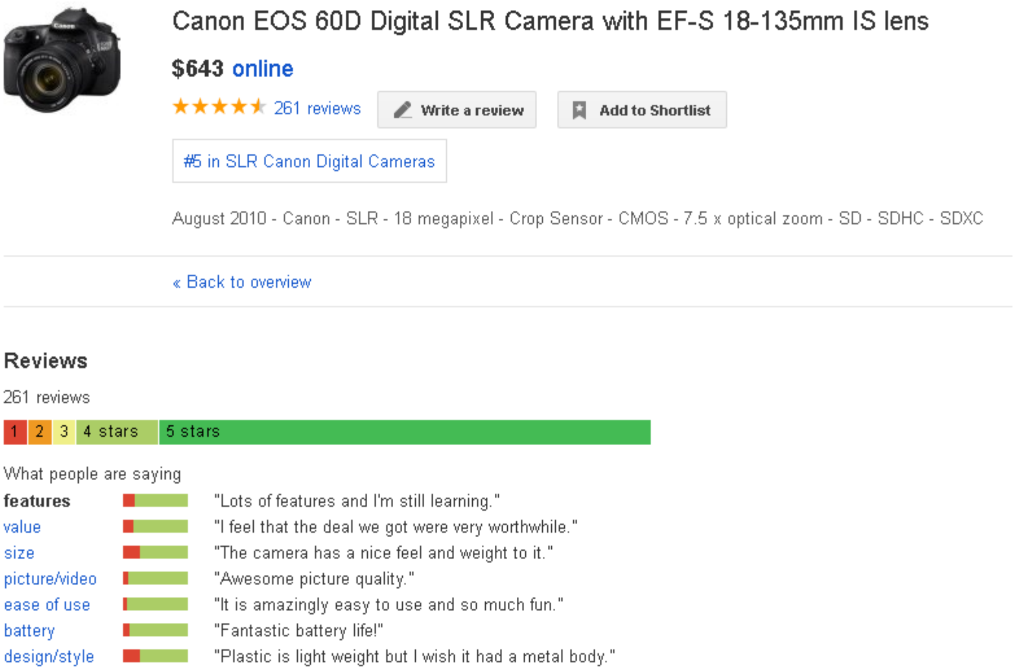
Cons: Lack of real manual white balance.

I own many Canon cameras/bodies, but the main to compare are my 40D, 20D and S95. Yes, the S95 vs. T3i. It is amazing.

When I bought this T3i on 12/31/11, I knew it had drawbacks.

Figure 3.2: Sample of reviews from Google Shopping

- Frequency and relation based approaches
  - Frequency-based approaches
  - Relation-based approaches
  - Hybrid approaches
- Model-based approaches
  - Supervised learning techniques
  - Topic modeling techniques



Canon EOS 60D Digital SLR Camera with EF-S 18-135mm IS lens

$643 online

★★★★★ 261 reviews

#5 in SLR Canon Digital Cameras

August 2010 - Canon - SLR - 18 megapixel - Crop Sensor - CMOS - 7.5 x optical zoom - SD - SDHC - SDXC

Reviews

261 reviews

What people are saying

features	"Lots of features and I'm still learning."
value	"I feel that the deal we got were very worthwhile."
size	"The camera has a nice feel and weight to it."
picture/video	"Awesome picture quality."
ease of use	"It is amazingly easy to use and so much fun."
battery	"Fantastic battery life!"
design/style	"Plastic is light weight but I wish it had a metal body."

Figure 3.3: Percentage of positive and negative sentences for each aspect at Google Shopping

In the following sections, we will discuss each of these categories, their strengths and weaknesses, and the state-of-the-art methods. Liu has also categorized rating prediction methods into two groups:

- Supervised learning techniques
- Lexicon-based approaches

For the supervised learning approach, the learning methods used for sentence-level sentiment classification have been applied at the phrase-level [173, 55, 15]. The main weakness of this approach is that it requires training data for each domain, and the model will depend on that training data. In other words, a model or classifier trained from labeled data in one domain often performs poorly in another domain. Although recently researchers have started working on domain adaptation techniques, this technology is still far from mature [92]. Another weakness of these techniques is that they need a lot of training data to perform an accurate classification.

Lexicon-based approaches use a sentiment lexicon, which contains a list of sentiments, to determine the orientation/rating of a given sentiment [30, 48, 117, 167]. This approach can avoid some of the issues of the supervised learning approach. Since these methods are typically unsupervised, there is no need for training data. It has also been shown that they perform quite well in a large number of domains [92]. The lexicon-based approach also has its own shortcomings: it is hard to use to find domain- or context-dependent orientations of sentiments. In other words, the sentiment orientations of words identified this way are domain- and context-independent [92]. However, there are many sentiments that have context dependent orientations, e.g., 'quiet' is negative for a speaker phone, but it is positive for a vacuum cleaner.

Since rating prediction is usually performed as a consecutive task of aspect identification, in this dissertation we will use the aspect extraction categorizations for grouping state-of-the-art techniques and will discuss the rating prediction method of each work in the description of that method.

3.2.1 Frequency- and Relation-based Approaches

Most of the early works on aspect-based opinion mining are frequency-based approaches [2, 3, 48, 54, 56, 93, 97, 132, 150, 195, 197]. These approaches usually apply some constraints on high-frequency noun phrases to identify aspects. Some of the early works are relation-based approaches and use the relationship between words to identify aspects. These methods usually employ a set of manually or automatically extracted POS patterns to extract aspects. Finally, there are some hybrid approaches that combine both frequency- and relation-based techniques to extract aspects. These methods normally apply relationship patterns for filtering the result of a frequency technique. In the following we review some of the most important frequency-based, relation-based and hybrid methods proposed in the literature.

Frequency-based Methods

Frequency-based methods usually apply a set of constraints on high-frequency noun phrases to identify aspects. An aspect can be expressed by a noun, adjective, verb or adverb. However, recent research [90] shows that 60-70% of the aspects are explicit nouns. In reviews people are more likely to talk about relevant aspects which suggests that aspects should be frequent nouns. However, not all of the frequent nouns are aspects. Therefore, different filtering techniques are applied on frequent nouns to filter out non-aspects.

- Strength: Although these methods are very simple, they are actually quite effective. Many companies are using these techniques for analyzing their user feedback.

- Limitations: These methods tend to produce too many non-aspects and miss low-frequency aspects. In addition, they require the manual tuning of various parameters (thresholds) which makes them hard to port to another dataset.

Feature-based Summarization (2004)

The method proposed in [48], called feature-based summarization (FBS), first determines all frequent noun phrases from full text reviews as candidate aspects. Then two pruning methods are applied to remove those candidate aspects whose words do not appear together in a specific order (meaningless) and those which are subsets of others (redundant).

The proposed sentiment orientation method is based on a set of 30 seed adjectives with known orientations (positive or negative). For each sentence in the reviews, if it contains an aspect, all the adjectives are extracted as (potential) sentiments. If a direct synonym or antonym of a sentiment is found in the seed set, its orientation is determined, otherwise the sentiment is ignored. The orientation of each sentence is derived from the sum of the orientations (+1 for positive and -1 for negative) of its sentiments. As output, for each discovered aspect the method shows the number of positive and negative opinion sentences.

OPINE (2005)

The OPINE method [132] first extracts all noun phrases from reviews and retains those with frequency greater than an experimentally set threshold. Each noun phrase is then evaluated by computing the Point-wise Mutual Information (PMI) between the phrase and associated discriminators. A discriminator consists of an extraction pattern with alternating context strings and slots (e.g., “great X”, “has X”, “comes with X” where X is the product aspect). Given a noun phrase f and discriminator d, the PMI score is defined as follows:

$$PMI(f,d) = \frac{Hits(d+f)}{Hits(d) \times Hits(f)} \tag{3.1}$$

OPINE applies an NLP parser to determine syntactic dependencies of words in each sentence and then generates a set of syntactic rules for extracting sentiment associated to each aspect. Finally, a classification technique is applied on the extracted sentiments to classify them as positive or negative.

Other Methods

There are also several other frequency-based techniques proposed in the literature. Ku et al. [75] present a method which makes use of the TF-IDF scheme considering terms at the document level and paragraph level. The method proposed by Scaffidi et al. [147] compares the frequency of extracted candidates (frequent noun phrases) in a review corpus with their occurrence rates in generic English. Zhu et al. [196] propose a method that considers the frequency of phrase t, the length of phrase t, and also other phrases that contain t. They first find a set of candidates, and then refine them using a bootstrapping technique with a set of given seed aspects. The idea of refinement is based on each candidate's co-occurrence with the seeds. Raju et al. [138] (making the same assumption as [147]) simply discard all the noun phrases that occur more frequently in general English than in product descriptions. The noun phrases are then clustered so that noun phrases describing the same aspect are grouped together in the same cluster. The model calculates bi-gram overlap to measure the similarity between two noun phrases. At the end, only clusters that contain at least three noun phrases are kept. Long et al. [97] extract noun phrases based on frequency and information distance. Their method first finds the core aspect words using the frequency-based method. It then uses the information distance to find other words related to an aspect, e.g., for aspect price, it finds '$' and 'dollars'. All these words are then used to select reviews which discuss a particular aspect.

Relation-based Methods

Relation-based methods exploit aspect-sentiment relationships to extract new aspects and sentiments. The intuition behind this approach is that each sentiment expresses an opinion on an aspect and sentiments are often known or easy-to-find [92]. As a result, their relationship can be used for identifying new aspects (and sentiments).

- Strength: Compared to frequency-based methods, relation-based approaches can find low-frequency aspects.
- Limitation: The main limitation of these methods is that they produce many non-aspects matching with the relation patterns.

Opinion Observer (2005)

Liu et al. [93] present a framework, called opinion observer, for the visual comparison of consumer opinions. This framework can only be used for short comments where aspects are commented on in sentence segments and there is no irrelevant information. Opinion observer uses a supervised pattern discovery algorithm to perform its tasks. The training dataset is first tagged using a POS tagger. Then actual aspects are manually identified and replaced by a specific tag, e.g., [aspect]. For example, the sentence segment 'good manual' is POS tagged as 'good_ADJ manual_NN' (where

ADJ and NN indicate adjective and noun respectively) and then manually tagged as 'good_ADJ [aspect]_NN'. Association rule mining is then applied to find POS patterns for extracting aspects.

Since not all of the generated patterns are useful, some conditions are defined to filter out patterns that are not sufficiently predictive. For sentiment orientation, they simply assume that sentiments appearing in a Pros section are positive while those appearing in a Cons section are negative.

Multi-Facet Rating (2009)

The method proposed in [3], called multi-facet rating, uses a set of predefined POS patterns to extract opinion phrases as candidate aspects. Then a lexical resource, called General Inquirer (GI) [155], is employed to predict the orientation of phrases and also to enrich candidate aspects with extra information. GI contains a set of adjectives labeled as positive or negative and a finer-grained set of sentiment-related tags, e.g., pain, feel, pleasure, emotion, strong, etc.

A filtering method, called minimum variance (MV), scores candidate aspects based on their discriminative power. The MV method is based on measuring the variance of distribution of a candidate aspect across the orientation label and retaining only those candidates that have the smallest variance.

Tree Kernel Approach (2010)

A simple method of using syntactic structures of the aspects is exact matching. This method that has been used in some of the previous works fails to handle similar syntactic structures and therefore cannot be generalized for unseen data. In [56], an approach based on tree kernels is proposed to address this limitation. Tree kernel based methods can be used to implicitly explore the substructure of the syntactic structure and calculate the similarity between two substructures. The proposed tree kernels in [56] encode not only syntactic structure information, but also sentiment related information, such as sentiment polarity.

Other Methods

In Zhuang et al. [197], a dependency parser is used to identify dependency relations for aspect extraction. Du et al. [32] utilize another type of relations between aspects and sentiments. They first consider all noun phrases as aspects and all adjectives as sentiments and then build a graph of aspects and sentiments based on their co-occurrence in reviews. A graph clustering algorithm (information reinforcement) is applied to find aspects highly related to sentiments. Hai et al. [43] utilize the co-occurrence matrix of aspect-sentiment for mining a set of rules for extraction of new pairs. In a similar work [136] the dependency idea is generalized into the double-propagation method for simultaneously extracting both sentiment words and aspects.

Hybrid Methods

Hybrid approaches combine both frequency- and relation-based techniques to extract aspects. These methods normally use aspect-sentiment relations for filtering frequent noun phrases. The intuition behind the hybrid approach is that aspects are mostly frequent nouns and are normally described by some sentiments. So, the relationship between aspects and sentiments can be used for filtering non-aspects.

- Strength: compared to frequency- and relation-based approaches, the number of non-aspects is more limited in the result of hybrid methods since they apply more constraints (frequency threshold and relation pattern).
- Limitation: While hybrid methods can limit the number of non-aspects, they still miss low-frequency ones. They also require the manual tuning of various parameters.

Li's Method (2009)

Li et al. [86] first extract frequent noun phrases from reviews as candidate aspects and then remove the noise through two filtering steps. The first filter is the adjective restriction pattern, i.e., $\_ADJ\_NN$. The second filtering step is based on the statistical characteristic of the Web text. The common words that occur too often on the Web text are regarded with low probability to is an aspect. So, they filter out a candidate if it is highly frequent in the background corpus. Note that in this work the whole web is considered as background corpus and TF is computed to get the frequency of a candidate in the background corpus.

Yu's Method (2011)

Yu et al. [183] propose a method to organize the product aspects into a hierarchy. The method first extracts frequent noun phrase from reviews and then train a one-class SVM to identify candidate aspects. Candidate aspects are then filtered out based on the inter-aspect semantic distance. The inter-aspect semantic distance is defined based a set of relationship features among aspects, e.g., co-occurrence, contextual similarity, syntactic pattern similarity, etc. After identifying the aspects, the method incrementally inserts them into the initial hierarchy which is automatically acquired from domain knowledge. The method finally performs a sentiment classification to determine consumer opinions on these aspects.

Other Methods

The method proposed in [190] adopts the double propagation idea to extract candidate aspects. Double propagation method exploits certain syntactic relations of sentiments and aspects, and propagates through both sentiments and aspects iteratively. Two improvements based on 'part-whole' relation patterns and a 'no' pattern are made to find candidates which double propagation cannot find. To filter out noises from extracted candidates, the authors propose a ranking algorithm to rank the extracted candidates based on two factors: candidate frequency and candidate correctness (the possibility that a candidate be a correct aspect).

3.2.2 Model-based Approaches

The major limitation of the frequency- and relation-based methods is that they require the manual tuning of various parameters which makes them hard to port to another dataset. Model-based techniques overcome this limitation by automatically learning the model parameters from the data.

Some of the proposed models are based on supervised learning techniques such as Hidden Markov Model (HMM) and Conditional Random Field (CRF). However, most of the current models are unsupervised topic models and based on Probabilistic Latent Semantic Indexing (PLSI) and Latent Dirichlet Allocation (LDA). In the following we review some of the most important and most recent models proposed in the literature for the problem of aspect-based opinion mining.

Supervised Learning Techniques

Aspect extraction can be seen as a special case of the general information extraction problem. Many algorithms based on supervised learning have been proposed in the past for information extraction. In aspect-based opinion mining, these methods can be applied on reviews to identify aspect, sentiments, and their polarity.

The most prominent methods for information extraction are based on sequential learning (or sequential labeling). The current state-of-the-art sequential learning methods are HMM and CRF. These methods infer a function from labeled (supervised) training data to apply to unlabeled data. HMM is a generative probabilistic model with two dependency assumptions: 1) The hidden variable at time t, namely y_t, depends only on the previous hidden variable y_{t-1} (Markov assumption). 2) The observable variable at time t, namely x_t, depends only on the hidden variable y_t at that time. The parameters are then learned by maximizing the joint probability distribution $p(x, y)$. CRF is a discriminative probabilistic model that can come in many different forms. The form that most closely resembles the HMM is known as a linear-chain CRF. The parameters of a CRF are learned by maximizing the conditional probability distribution $p(y|x)$.

- Strength: supervised learning approaches overcome the limitations of frequency- and relation-based methods by learning the model parameters from the data.

- Limitation: These models need manually labeled data for training.

Wong's Model (2008)

Wong et al. [178] propose a probabilistic graphical model for jointly extracting and grouping aspects from multiple Websites. Their method considers the page-independent content information and the page-dependent layout information in a single framework. This model is integrated with an HMM component to generate the content information. They assume that each text fragment consists of an individual HMM which generates that fragment.

The generative process of this model is as follows: suppose we have a collection of N different text fragments collected from S different Web sites. A text fragment refers to the text unit displayed in a Web browser. Each generation of a text fragment is modeled as an independent and identical event. For generating the nth text fragment, a topic Z_n is first sampled. Next, the indicator T_n is generated, representing whether the topic is an aspect or not. Then for that website the layout information L_s is generated. Finally, an HMM is employed to generate content. The model predicts the label of each token as attribute-name, attribute-value or attribute-irrelevant.

OpinionMiner (2009)

The model proposed in [58], called OpinionMiner, is based on HMM. The main tasks of this model are identifying aspects, sentiments, and their polarity. The novelty of this work is integrating POS information with the lexicalization technique. In other words, the model integrates POS information in the HMM framework, i.e., the generation of each word depends not only on its previous word, but also on its part of speech tag.

The authors first define a set of tags: aspect, explicit positive, explicit negative, implicit positive, implicit negative, and background word. All aspects, sentiments, and sentiment polarities in a set of documents are manually labeled using the defined tag set. A bootstrapping approach is then applied on the labeled documents to enhance the training set. Finally, an HMM-based model is applied to extract aspects and opinion expressions.

Skip-Tree CRF (2010)

The authors of [81] propose a series of CRF models for extracting aspects, related sentiments, and the polarity of sentiments from reviews. Besides the neighbor context modeled by linear-chain CRF, they propose to use Skip-chain CRF and Tree CRF to utilize the conjunction structure and syntactic tree structure of the sentence. The Skip-chain CRF model assumes that if words or phrases are connected by the conjunction 'and', they mostly belong to the same opinion polarity. It makes the reverse assumption for words connected by the conjunction 'but'. Tree-chain CRF considers the syntactic tree structure of reviews which provide deeper syntactic dependencies for aspects and sentiments. A unified model, called Skip-Tree CRF, is proposed to integrate these two structures.

Other Methods

Sauper et al. [146] combine topic modeling with HMM, where the HMM models the sequence of words with types (aspect word, sentiment word, or background word). The authors of [54, 149] propose CRF-based models. In [54] a CRF model is trained on review sentences from different domains for a more domain independent extraction. A set of domain independent features are also used, e.g. tokens, POS tags. CRF is also used in Choi [22]. In this work, a set of sequential pattern rules are mined using a sequential pattern mining technique considering labels (or classes), dependency, word distance, and opinion sentences.

Topic Modeling Techniques

Topic modeling is an unsupervised learning method that assumes each document consists of a mixture of topics and each topic is a probability distribution over words. A topic model is basically a generative model which specifies a probabilistic procedure by which documents can be generated. The output of topic modeling is a set of word clusters. Each cluster forms a topic and is associated with a probability distribution over words in the document collection. Topics from topic models can be considered as aspects. Topic modeling can thus be applied to extract aspects. However, there is also a difference, i.e., topics can cover both aspects and sentiments. For aspect-based opinion mining, they need to be separated. This separation can be achieved by extending the basic topic models to jointly model both aspects and sentiments [92].

There are two main basic topic models: Probabilistic Latent Semantic Indexing (PLSI) [47] and Latent Dirichlet Allocation (LDA) [14]. PLSI is a statistical technique for the analysis of co-occurrence data. LDA is similar to PLSA, except that in LDA the topic distribution is assumed to have a Dirichlet prior which results in more reasonable mixtures of topics in a document. In both models documents are represented as mixtures over latent topics and topics are associated with a distribution over the words of the vocabulary.

- Strengths: In contrast to supervised learning models, there is no need for manually labeled data. In addition, topic models perform both aspect extraction and grouping at the same time in an unsupervised manner.

- Limitation: Topic models normally need a large volume of (unlabeled) data to be trained accurately.

TSM (2007)

A probabilistic model is presented in [104] to identify aspects and their polarity simultaneously. The proposed model is called Topic-Sentiment Mixture (TSM) and is based on PLSI. The novelty of this model is utilizing the distribution of background words. In this model a document is assumed to be generated by sampling words from a set of topic distributions and two (positive and negative) sentiment distributions.

According to this model, it is first decided whether the word will be a common English word. If so, the word is sampled from the distribution of background words. If not, it is decided which of the k aspect distributions the word is sampled from. Finally, it is decided whether the word is used to describe the topic neutrally, positively, or negatively.

MG-LDA (2008)

Titov et al. [163] propose a topic model, based on LDA, for extracting aspects from reviews. Their topic model extracts two types of topics from reviews: global topics and local topics. They assume global topics correspond to a global property of the product in the review, such as its brand, and local topics correlate with the product aspects. They assume the distribution of global topics is fixed for a document, but the distribution of local topics is allowed to vary across documents. They use an existing ranking algorithm to estimate the rating of known aspects based the learned feature vectors.

In [164] the same authors extend their model to find the correspondence between the extracted topics and the product aspects. The authors assume that the rating of a known aspect is correlated to the words used for describing that aspect in the review. Therefore, the model is extended through a set of maximum entropy classifiers, one per known aspect, that are used to predict their ratings. The model enforces that only words assigned to an aspect's topic are used in predicting the rating of that aspect. The authors conclude that representing a document as a mixture of latent topics which generate all words of the document, is not a good choice for the problem of aspect-based opinion mining since the extracted topics are very general.

Lu's Model (2009)

Based on the conclusion of [164], Lu et al. [101] assume that each short comment can be parsed into an opinion phrase as a pair <head term, modifier>. Comments are pre-processed by chunking them into opinion phrases, and then models are learned that generate only opinion phrases, not all the words of a review. The authors propose a probabilistic model based on PLSI to identify major aspects of a product by clustering the head terms. They assume that if two head terms use the same set of sentiments, they should share similar meaning. The orientation of a head term is considered the same as the polarity of the corresponding short comment which can be positive or negative. The orientation of each cluster is then computed by aggregating the polarity of all head terms in that cluster.

Guo's Model (2009)

Guo et al. [42] propose an unsupervised method based on latent semantic association (LaSA) for grouping candidate aspects. They assume that words in the same context have similar semantic association. In the first step, all noun phrases from the semi-structured Pros and Cons are extracted as candidate aspects. The extracted candidates are then verified by checking their re-occurrence in the full text reviews. In the second step, two alternative LaSA models are used to group the extracted aspects. The first LaSA model employs the page-independent context information to group aspects. This model assumes that instances of an aspect behave in the same way and thus groups words into a set of aspects according to their context in the reviews. Given a word t, its context is defined to be composed of all the sentiments adjacent to t in the corpus. The second model uses page-dependent layout information of text fragments for grouping candidate aspects. This model groups words according to their latent semantic structures. Given a latent aspect a, its semantic structure is defined to be composed of the context units of all the words generated by a.

MaxEnt-LDA Model (2010)

Another LDA-based model for jointly identifying aspects and sentiments is proposed in [194]. The novelty of this model is the integration of a discriminative maximum entropy (Max-Ent) component with the standard generative component. The Max-Ent component allows the model to leverage POS tags of words to help separate aspects, opinions, and background words. The authors assume there are two types of opinions in a review: aspect-specific opinion words which are each associated with only a single aspect (e.g., 'tasty' and 'friendly' which are associated with 'food' and 'staff', respectively), and general opinion words which are shared across different aspect (e.g., 'great' in sentence 'The food was great!'). They introduce two indicator variables to distinguish between aspects, opinions, and commonly used background words.

Brody's Model (2010)

Brody et al. [19] apply the LDA model at the sentence level to extract local topic as aspects. For each aspect, the relevant adjective is then extracted as sentiment and a conjunction graph is built over adjectives. Using a set of seed adjectives with known polarity, the polarity of other adjectives is determined by propagating the polarity scores across the graph.

ASUM Model (2011)

In a work similar to [19], Jo et al. [64] assume all words in a single sentence are generated from one aspect and apply LDA at the sentence level to extract aspects. This model is further extended to extract sentiments related to each aspect. In this model, each review has a distribution over sentiments and each sentiment has a distribution over aspects. To generate a review, the model first draws the document's sentiment distribution θ_d. To generate each sentence i, a sentiment s_i is first selected from θ_d and then an aspect a_i is chosen conditional on s_i. Each word of the sentence i is then generated based on a_i and s_i.

LARAM Model (2011)

The authors of [171] propose an LDA-based model to jointly identify not only aspects and their ratings, but also the weight placed on each aspect by the reviewer. The model takes a set of reviews about a product with overall ratings assigned by reviewers to that product. The generative assumption of a reviewer's rating behavior in this model is as follows: the reviewer first decides the set of aspects he wants to comment on, and then for each aspect, the reviewer chooses the words with appropriate sentiment polarities to reflect his opinions on the aspects. Finally, the reviewer assigns an overall rating based on the weighted sum of all the aspect ratings.

CFACTS (2011)

Lakkaraju et al. [77] propose a probabilistic model, called CFACTS, to jointly discover latent aspects and sentiments, and also estimate the rating of each sentiment. The proposed model combines topic modeling with HMM to capture the syntactic dependencies between the aspects and sentiments. The model assumes that each review has a distribution over aspects, and another distribution over sentiments. To generate the ith word of a document, aspect a_i and a sentiment s_i are chosen from the corresponding distributions. The syntactic class of the word which determines whether the word is an aspect, a sentiment, or a background word, is chosen based on the syntactic class of the previous generated word.

The proposed model also incorporates coherence in reviews for more accurate aspect and sentiment discovery. Coherence in a review means that the reviewer keeps talking about one aspect or sentiment before moving to another topic. A window, which is a contiguous sequence of words, is considered as the basic unit of coherence. All words within a window are assumed to be derived from the same topic (aspect or sentiment). Finally, the ratings of sentiments are computed using a learned normal linear regression model.

STM (2011)

Lu et al. [99] present a Segmented Topic Model (STM) to identify topics as aspects. The novelty of this model is jointly modeling document-level and sentence-level topics. The generative process of this model is as follows: For each document a document-level topic distribution is drawn. Then for each sentence, a sentence-level topic distribution is selected. Finally, for each word in a sentence, a topic is selected for generating that word (as an aspect). The authors proposed a separate model for rating prediction, training a regression model on overall ratings and applying the model on the corresponding aspect.

SDWP Model (2011)

Similar to [101], Zhan's model [188] is defined on opinion phrases. They employ a dependency parser on review texts to extract these phrases. Then the LDA model is applied on the extracted opinion phrases to cluster them into k groups. Head terms and modifiers are clustered together, i.e., each cluster contains both head terms and modifiers.

JST (2011)

The authors of [46] assume that each document has a specific distribution over the polarity of words. They further assume that there is a specific topic distribution for each polarity (positive and negative) which is independent from the documents. The proposed model first chooses a polarity label l from the document specific polarity distribution and then selects a topic randomly from the topic distribution conditional on the polarity label l. They use the learned model for classifying movie reviews as positive or negative. The authors further modify the model by incorporating word polarity priors through modifying the word Dirichlet priors.

Other Models

Branavan et al. [17] propose a method which makes use of the aspect descriptions as key phrases in Pros and Cons to help finding aspects in the full text review. This model consists of two parts. The first part clusters phrases in Pros/Cons into some aspect categories based on distributional similarity. The second part builds a topic model modeling the aspects in the review text. In Mukherjee et al. [118], a semi-supervised joint model is proposed, which allows the user to provide some seed candidate aspects for some aspects in order to guide the inference to produce aspect distributions that conform to the user's need.

There are also some works [88, 46] proposing LDA-based models for extracting topics and their polarity from documents. While these works do not talk about aspects and rating specifically, their models can be applied on reviews for identifying aspects and their ratings. The authors of [88, 46] assume that each document has a specific distribution over the polarity of words and that there is a specific topic distribution for each polarity (positive and negative) which is independent from the documents (global topic distribution). To generate a word, the proposed model in [82] first chooses the polarity of word form the document specific polarity distribution. Then it selects a topic from the global topic distribution conditioned on the selected polarity. Finally, a word is generated conditioned on the selected topic and polarity.

3.3 Challenges

There are various challenges that make the problem of aspect-based opinion mining hard. In the following we list some of them:

The first challenge in identifying aspects is that different reviewers may use different words or phrases to express the same aspect, e.g.,

- *Photo quality* is a little better than most of the cameras in this class.
- That gives the SX40 better *image quality*, especially in low light, experts say.
- These *images* are recorded in full resolution, making it particularly useful for shooting fast moving subjects.

Likewise, different reviewers tend to use different sentiments for expressing the same rating, e.g.,

- For a camera of this price, the picture quality is *amazing*.
- I am going on a trip to France and wanted something that could take *stunning* pictures with, but didn't cost a small fortune.

Another challenge is noisy information. Full text reviews normally include a large amount of irrelevant information, e.g., opinion about the manufacturer of the product and information about the reviewer.

- Canon is a company that never rests on its laurels, instead choosing to make continuous refinements and upgrades to its cameras.
- I have owned Canon power shot pocket cameras exclusively over the years.
- I have fat hands but short fingers.

While explicit aspect/sentiment extraction has been studied extensively, limited research has been done on extracting implicit ones. However, there are many aspects/sentiments in reviews which are understandable for a human reader but hard to be extracted by a machine, e.g.,

- After a twenty-one mile bike ride a four mile backpacking river hike, the size, weight, and performance of this camera *has been the answer to my needs*.
- The grip and *weight make it easy to handle* and the mid zoom pictures have exceeded expectation.

Identifying opinions in comparative sentences is also very challenging. A comparative opinion expresses a relation between two or more items and/or a preference of the reviewer based on some shared aspects of the items, e.g.,

- This camera is everything the SX30 should have been and was not.
- The SX40 HS significantly improves the low light performance of its predecessors.
- Yes, I have used comparable Nikon and Olympus products. The SX40 HS is the best for me...
- I bought this camera as an upgrade to my Panasonic Fz28 which I have had for a few years and found it to be inferior in almost every aspect.

The last challenge that we want to discuss here is co-reference resolution. Co-reference occurs when multiple expressions in a sentence or document refer to the same thing, e.g.,

- The batteries are great. *They* last 10 hours.
- The new LI-ion battery is a good fit and *it* charges fast inside of three hours.

3.4 Evaluation Metrics

If the necessary ground truth is available, the performance of a method for aspect-based opinion mining can be evaluated by measures such as accuracy, precision and recall. However, in real-life data sets such ground truth is typically not available. In some of the works some human judges have been asked to read a set of reviews and manually create a set of "true" aspects and their ratings for the reviewed item as "gold standard". Precision and recall of aspect extraction are then computed versus this gold standard. Precision and recall are defined as follows:

$$Precision = \frac{|ExtractedAspects \bigcap TrueAspects|}{|ExtractedAspects|} \tag{3.2}$$

$$Recall = \frac{|ExtractedAspects \bigcap TrueAspects|}{|TrueAspects|} \tag{3.3}$$

Sometimes F-measure is also used for evaluation of extracted aspects:

$$F - measure = \frac{2 \times Recall \times Precision}{(Recall + Precision)} \tag{3.4}$$

To evaluate the accuracy of aspect ratings, the estimated ratings for the extracted aspects are compared with their gold standard ratings. Mean Absolute Error (MAE), Mean Squared Error (MSE), and Ranking loss are commonly used for evaluation of the estimated ratings:

$$MAE = \frac{1}{k}\sum_{i=1}^{k} |\hat{r_i} - r_i| \tag{3.5}$$

$$MSE = \frac{1}{k}\sum_{i=1}^{k} (\hat{r_i} - r_i)^2 \tag{3.6}$$

where $\hat{r_i}$ is the estimated and r_i is the true rating of the ith aspect, and k is total number of aspects. Ranking Loss measures the average distance between the true and predicted rating of an aspect over different items. Overall ranking loss is simply the average over each aspect. Given N test instances, the ranking loss for an aspect is equal to:

$$RankingLoss = \sum_{n} \frac{|ActualRating_n - PredictedRating_n|}{N} \tag{3.7}$$

While the aforementioned metrics are meaningful from an application point of view, obtaining a ground truth is normally expensive as it typically requires manual labeling. Therefore, topic modeling approaches often use a standard approach for the evaluation in the absence of the ground truth. They perform cross-validation learning the model from a training set and computing the likelihood of a held-out test set. Normally 10% to 20% of the reviews are held out for testing purposes and the remaining are used to train the model. Perplexity is one of the standard measures of evaluating these models. The perplexity is monotonically decreasing in the likelihood of the test data, and a lower perplexity score indicates better performance. More formally, for a test set of N reviews, the perplexity is defined as [14]:

$$perplexity(\boldsymbol{D}_{test}) = exp\{-\frac{\sum_{d=1}^{D} \log P(\boldsymbol{w}_d)}{\sum_{d=1}^{D} N_d}\} \tag{3.8}$$

where $P(\boldsymbol{w}_d)$ is the likelihood of test document d. While there is no need for ground truth to evaluate based on this metric, it is less meaningful from an application point of view.

3.5 Benchmark Data Sets

In this section we briefly explain some of the publicly available review data sets:

Labeled review datasets

There are two public datasets with ground truth. The Amazon labeled dataset[2] contains 314 reviews about 5 products. In this dataset aspects in reviews are manually tagged. A rating in the range from -3 to +3 is also assigned to each identified aspect.

We also crawled the Epinions website and prepare a larger dataset with ground truth[3]. This dataset contains 2483 reviews in 5 product categories: camcorder, cellular phone, digital camera, DVD player, and MP3 player. There are 8 products in each category with different overall ratings. This dataset has the following information for each review: title, product overall rating, list of known aspects and their ratings, Pros and Cons, and full text review. This dataset contains a set of true aspect for each item, but the ratings of aspects are not provided.

Hotel review dataset from TripAdvisor [4]

In the TripAdvisor dataset there are 37,181 reviews about 2,232 hotels written by 34,187 reviewers. The average length of each review in this dataset is 96.5 words. In addition to the overall ratings, reviewers are also asked to provide ratings on 7 known aspects in each review (value, room, location, cleanliness, check in/front desk, service, business service) ranging from 1 star to 5 stars.

MP3 review dataset from Amazon [5]

The MP3 dataset of Amazon contains 16,680 reviews about 686 MP3 players written by 15,004 reviewers. The average length of each review in this data set is 87.3 words. There is only one overall rating in each review, ranging from 1 star to 5 stars.

[2] http://www.cs.uic.edu/~liub/FBS/CustomerReviewData.zip

[3] http://www.sfu.ca/~sam39/Datasets/LabeledEpinions/

[4] http://sifaka.cs.uiuc.edu/~wang296/Data/LARA/TripAdvisor/

[5] http://sifaka.cs.uiuc.edu/~wang296/Data/LARA/Amazon/mp3/

Review dataset from Amazon [6]

The Amazon data set contains 5.8 million reviews from 2.14 million reviewers. Each review consists of 8 parts: product Id, reviewer Id, rating, date, review title, review text, number of helpful feedbacks, number of feedbacks. This complete data set can be used for different opinion mining tasks such as opinion helpfulness prediction.

Review dataset from Epinions [7]

The last benchmark dataset is a review dataset that we crawled from Epinions. This data set contains 1.5 million reviews about 200K products written by 326K reviewers. This data set contains not only text of reviews, but also helpfulness votes of reviews assigned by different users (raters). There are 120K raters, 755K rated reviews and 13 million ratings in this data set.

[6] `http://liu.cs.uic.edu/download/data/`

[7] `http://www.sfu.ca/~sam39/Datasets/EpinionsReviews/`

Chapter 4

Opinion Digger: A Hybrid Method for Mining Reviews

In this research we propose a hybrid method, called *Opinion Digger*, to mine and summarize opinions from customer reviews [109]. Opinion Digger takes review texts and supervision data as input, and outputs a set of additional aspects (not provided in input), plus the estimated rating of each.

In most reviewing websites such as Epinions.com four types of supervision data are available for each review in addition to the text: known aspects and their ratings, a rating guideline, Pros and Cons comments, and the overall rating of the reviewed item. Some of the existing opinion mining methods have used part of the supervision available in reviews. For example, to determine the orientation of sentiments and aspects, [101] used the overall ratings of reviews and [93] used the Pros and Cons labels.

We conduct experiments on a real life dataset from Epinions.com, demonstrating the improved effectiveness of Opinion Digger in terms of accuracy of aspect extraction. Evaluation of rating estimation and candidate grouping also demonstrate the high accuracy of the proposed techniques.

This chapter is organized as follows: In the next section, we describe our contributions and briefly explain different phases of Opinion Digger. Sections 4.2 and 4.3 present aspect extraction and rating prediction methods of Opinion Digger, respectively. In Section 4.4 we report the results of our experimental evaluation on a dataset from Epinions.com. Finally, Section 4.5 concludes this chapter with a summary and the discussion of future work.

4.1 Our Contributions

As discussed in chapter 3, most of the early works on aspect-based opinion mining are frequency-based approaches. Frequency-based approach provides a good set of candidate aspects that needs to be filtered to get actual ones. Relation-based approaches use the aspect-sentiment relationships to identify aspects and sentiments. One of the relationships that is mainly used is the syntactic relation between aspects and sentiments. In this work, we take advantages of both approaches and propose a method, called Opinion Digger, for identifying aspects and predicting their rating. A simple way to merge these approaches is to use a set of predefined syntactic patterns for filtering. However, syntactic patterns can only be used for the language and the type of text (full sentences, sentence segments, phrases, etc.) they are defined for. In other words, each language or text type has its own grammatical structure and therefore syntactic patterns. To this end, Opinion Digger first mines a set of opinion patterns from the given text (review) and then uses it for filtering frequent noun phrases.

While one of the important phases of aspect-based opinion mining is grouping synonymous candidate aspects (e.g., LCD, display, screen), it has been ignored in most of the previous works. Opinion Digger also uses a novel method to group synonymous candidates and select a representative for each group as the aspect. In addition, in contrast to the previous methods where mostly determine whether an opinion is positive or negative, Opinion Digger determines the strength of positiveness and negativeness of each aspect by assigning a numerical rating in the range from 1 to 5. To clarify the importance of opinion strength consider the following examples: “The picture quality is OK”, and ‘It has extraordinary picture quality”. Both sentences about the picture quality of two cameras are positive, but if you want to buy one based on just these two opinions, you will go for the second one.

In the following we briefly explain how our method uses the available supervision data. Opinion Digger uses known aspects to mine opinion patterns from reviews, and also to determine two thresholds in the filtering and grouping phases. The rating guideline is employed to estimate the rating of sentiments and therefore aspects. The overall ratings of reviews and the ratings of known aspects are withheld from the learning method and are only used for evaluation purposes. Note that Opinion Digger cannot exploit these types of semi-supervision, since it distinguishes not only two different orientations, but numeric ratings on a scale from 1 to 5.

4.2 Aspect Extraction

As discussed in the previous chapter, recent research [90] shows that 60-70% of the aspects are explicit nouns. In addition, aspects are more likely to be discussed by people which suggests that aspects should be "frequent nouns" [48]. Opinion Digger makes the same assumption and extracts frequent noun phrases as candidate aspects. However, not all of the frequent nouns are aspects. So, in the second step Opinion Digger mines a set of patterns for filtering out non-aspects. Finally, in the last step it groups synonymous candidates and selects one representative for each group as the identified aspect. The aspect extraction phase of Opinion Digger is shown in Figure 4.1. In the following subsections, we discuss each of these phases using real examples.

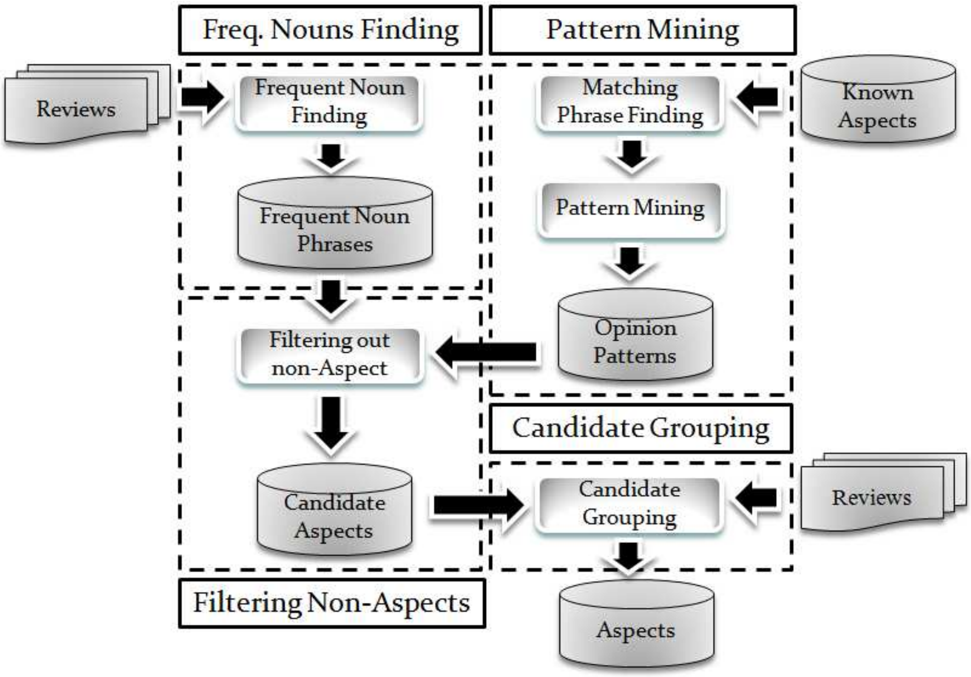


Figure 4.1: Aspect extraction phase

4.2.1 Finding Frequent Noun phrases

Opinion Digger first finds frequent noun phrases as potential aspects. It performs Part-Of-Speech (POS) tagging on the collection of reviews to determine the POS tag of each word (i.e., to determine whether the word is a noun, verb, adjective, etc). Opinion Digger uses the 'pos_tagger' which is a built-in POS tagger in NLTK[1], to generate the POS tag of each word. For example, the sentence segment "has inaccurate screen" is tagged as "has_VB inaccurate_JJ screen_NN" where _NN indicates a noun, _VB a verb, and _JJ an adjective.

After determining the POS tag of all words, Opinion Digger finds the stem of each noun (_NN) using the Porter Stemmer algorithm [133]. It eliminates all stop words (using stopword lists provided by Salton and Buckley [2], [3]) and only keeps nouns with non-stopword stems. Then Opinion Digger applies a modified version of Apriori [1] algorithm on the remaining nouns to find all multi-part noun phrases which are frequent, e.g., photo quality and LCD display. Apriori is a classic algorithm for frequent itemset mining. We modify the algorithm so that the position of words in the sentences are considered (frequent phrase mining). The support of each phrase in this algorithm is equal to the number of times it appears in the review collection. In our work, we use the minimum support of 1% to find frequent noun phrases as it is used in [48] and [49] for the same purpose.

Table 4.1 shows the top 10 frequent nouns for some products in descending order of frequency. As it is shown, there are some frequent phrases in each list which are not real aspects. Non-aspects are distinguished with gray color, like canon and elura for $Camcorder_i$.

Table 4.1: The top 10 frequent noun phrases for some sample products

$Camcorder_i$	$CellPhone_j$	$DigitalCamera_k$	$DVDPlayer_m$	$MP3Player_n$
camera	phone	camera	player	player
camcorder	nokia	picture	dvd player	pocket
video	aspect	battery	dvd	intel
light	time	photo	movie	concert
quality	battery	quality	toshiba	memory
canon	message	olympus	aspect	song
mode	address	card	picture	battery
picture	text	aspect	quality	headphone
elura	software	price	sound	unit
battery	data	time	output	time

We observed that not all of the known aspects for a product appear in the candidate aspect lists.

[1] http://www.nltk.org/

[2] http://www.lextek.com/manuals/onix/stopwords1.html

[3] http://www.lextek.com/manuals/onix/stopwords2.html

For example, portability and durability are known aspects for $MP3Player_n$, but they are in the list of frequent noun phrases of this product. We checked some of the reviews to find the reason, and it seems that in some cases, reviewers use different words (synonyms), to refer to the same aspect. For example, instead of portability reviewers preferred to talk about size of the product which is a potential aspect. One of the important aspects of each product is durability. However, most reviews were written in a time period close to purchase time, and therefore reviewers more likely to talk about other aspects than durability. On the other hand, some non-aspects are among the list of frequent noun phrases, so that we cannot rely on this method alone. In the following we will explain how Opinion Digger makes use of known aspects to mine patterns matching actual aspects and how to exploit these patterns to filter out non-aspects.

4.2.2 Mining Opinion Patterns

In this phase Opinion Digger uses known aspects and mines a set of POS patterns they match. We emphasize that mined patterns are independent from products, so the method learns the patterns across all reviews. In addition, opinion patterns will depend on the types of reviews, therefore if they are mined from short comments (or full text review), they can be applied to short comments (or full text review) to extract aspects.

To mine patterns, Opinion Digger first finds matching phrases for each of the known aspects. It searches for each known aspect in the full text reviews and finds its nearest adjective in that sentence segment as corresponding sentiment. It saves the sentence segment between these two as a matching phrase and picks the POS tags of all words as a pattern. It replaces the tag of known aspects with the special tag '_ASP' to identify which part of patterns are aspects. For example, one of the mined patterns using the known aspect 'movie quality' is '_JJ_ASP' which was extracted from "It has great movie quality".

After mining all POS patterns, the system uses Generalized Sequential Pattern (GSP) mining [154] to find frequent patterns. GSP is an algorithm used for sequence mining. We use 1% as the minimum support as it is used in [93]. Table 4.2 shows the most frequent mined patterns and some of the sentence segments they are extracted from. Note that these patterns are generic and independent from the products. In this table _ASP indicates an aspect, _NP a noun phrase, _JJ an adjective, _VB a verb, _IN a preposition, and _CC a coordinating conjunction. Determiners and adverbs are not considered in mining and also matching of patterns, since nouns can come with or without determiners and adjectives can come with or without adverbs.

Table 4.2: Frequent opinion patterns

Mined Patterns	Examples
_JJ_ASP	disappointing picture quality
_ASP_VB_JJ	sound is great
_ASP_IN_JJ	movie quality as good (as)
_ASP_IN_NP_VB_JJ	sound quality of this unit is amazing
_JJ_NP_IN_ASP	astonishing hours of battery life
_JJ_IN_ASP	unimpressed with photo quality
_JJ_NP_VB_ASP	lovely aspect is sound
_JJ_ASP_CC_ASP	great image and sound
_ASP_VB_VB_JJ	battery life has been good

4.2.3 Filtering Out Non-Aspects

In addition to the frequency threshold, we also put another constraint on potential aspects to filter out non-aspects and increase the accuracy of aspect extraction. In this work we employ a simple constraint on the number of matching patterns. We define the factor $Pnum$ which is the number of opinion patterns that are matched at least once by the potential aspect. Since the average value of $Pnum$ for known aspects is 2, a frequent noun phrase will be filtered out if $Pnum < 2$. After applying this constraint, Opinion Digger outputs a list of candidate aspects to the next phase. Table 4.3 shows the top 10 candidate aspects in descending order of $Pnum$ (again non-aspects are distinguished with gray color).

Comparing Tables 4.1 and 4.3 indicates that applying opinion patterns can eliminate most of non-aspects from the set of frequent noun phrases. However, some of them still remain since they match the minimum number of patterns. For example, headphone, concert and intel are some candidate aspects for the product $MP3Player_n$. The candidate 'headphone' conforms to enough patterns which correctly indicates that it is a real aspect. On the other hand, 'concert' cannot pass this constraint and so it is considered as non-aspect. 'intel' also can pass the pattern constraint and so it is falsely considered as a real aspect.

4.2.4 Grouping Candidate Aspects

Since different people use different words or phrases to express the same aspect, grouping synonyms helps reducing the size of the extracted aspect set. While most of the previous methods do not

Table 4.3: The top 10 candidate aspects for products in Table 4.1

$Camcorder_i$	$CellPhone_j$	$DigitalCamera_k$	$DVDPlayer_m$	$MP3Player_n$
camera	phone	camera	Player	headphone
camcorder	nokia	picture	price	button
light	battery	quality	button	memory
video	function	photo	dvd	player
joystick	wap	battery	movie	song
mode	button	price	picture	intel
quality	calendar	screen	sound	music
picture	model	shot	control	package
hand	note	card	disc	sound
size	option	mode	money	battery

consider aspect grouping at all, some of them used simple idea of grouping synonyms. For example, the authors of [93] employ a lexical dictionary to group synonymous aspects. Although selecting a good representative aspect for each group is very important, they do not mention how they select representatives.

In this work, we propose a novel technique based on Maximal Marginal Relevance (MMR) [20] to group synonymous aspects and select the best representative for each group. The MMR method is used for ranking documents in information retrieval and attempts to maximize the relevance while minimizing the similarity. To find representative aspects, A_i, for item i from the candidate aspect set C_i, in each step our candidate grouping technique computes the MMR values of all candidates in C_i and the candidate with the highest MMR value, say c_{in}, is selected. Opinion Digger puts c_{in} in the group with the representative aspect a_{im}, if the similarity between c_{in} and a_{im} is more than a threshold and more than the similarity to other representative aspects, otherwise it will create a new group and add c_{in} as its representative aspect. In both cases c_{in} will be removed from C_i. Note that, the representative aspect set of each item is empty at the beginning $A_i = \{\}$. The MMR values for candidate aspects C_i and representative aspects A_i is computed as follows:

$$MMR(A_i, C_i) = \arg\max_{c_{ij} \in C_i} [\lambda NLF(c_{ij}, i) - (1 - \lambda) \max_{a_{it} \in A_i} Sim(c_{ij}, a_{it})] \tag{4.1}$$

where $NFL(.)$ computes the relevance score and $Sim(.)$ computes the similarity score. λ is a parameter to adjust the combined score to emphasize the relevance ($\lambda = 1$) or to avoid redundancy

($\lambda = 0$). In our experiments, we set the λ to a value that selects the most number of known aspects as representatives and that is $\lambda = 0.5$. Setting the λ to higher and lower values does not qualitatively change the results. $NLF(c_{ij}, i)$ is the normalized logarithmic frequency [102] of candidate c_{ij} which indicates the relevance of the candidate c_{ij} to the collection of reviews available for item i, D_i:

$$NLF(c_{ij}, D_i) = \frac{1 + log(TF_{c_{ij},D_i})}{\sqrt{\sum_{t \in D_i}(1 + log(TF_{t,D_i}))^2}} \tag{4.2}$$

where D_i is the set of reviews available for item i, and TF_{c_{ij},D_i} is the frequency of candidate aspect c_{ij} in the given review set. The intuition behind using this value is that, an aspect will be a good representative if it has been used by more reviewers. We do not consider the inverse document frequency (idf), since some of the aspects are common between different categories (e.g. battery life, size, etc.). $Sim(c_{ij}, a_{it})$ computes the similarity between two given aspects by computing the shortest path distance between them in the synonymy graph of Wordnet [35].

$$Sim(c_{ij}, a_{it}) = 1/distance(c_{ij}, a_{it}) \tag{4.3}$$

As discussed above, in each step we pick the c_{ij} which has the highest MMR value. We experimentally set the group similarity threshold to $\alpha = 0.5$ to construct aspect groups. According to this threshold, words with the maximum distance 2 in Wordnet are grouped together. So, the aspect grouping method can be formalized as below:

$$\begin{cases} \forall a_{it} \in A_i, Sim(c_{ij}, a_{it}) < \alpha \Rightarrow A_i = A_i \bigcup \{c_{ij}\}, C_i = C_i - \{c_{ij}\} \\ \exists a_{ik} \in A_i, Sim(c_{ij}, a_{ik}) \geq \alpha, \forall a_{it} \in A_i, Sim(c_{ij}, a_{it}) < Sim(c_{ij}, a_{ik}) \\ \Rightarrow Group(a_{ik}) = Group(a_{ik}) \bigcup \{c_{ij}\}, C_i = C_i - \{c_{ij}\} \end{cases}$$

Table 4.4 shows some aspect groups and the selected representatives for MP3 Player category.

4.3 Rating Prediction

The second task of Opinion Digger is rating prediction (Figure 4.2). To the best of our knowledge, all of the previous works just consider two orientations for an opinion, positive and negative, but they do not express the strength of positiveness or negativeness of an opinion. In other words, they do not clarify how much an opinion is positive/negative and to what extent a reviewer recommends/not

Table 4.4: Some candidate groups and selected representatives for the $MP3Player_n$

Representative aspect	Candidates		
screen	screen	lcd	
accessory			
software	software	application	
sound	sound	music	song
headphone			
battery life	battery life	battery	life

recommends that product to others. In this work we consider a 5-level orientation scale (in the range from 1 to 5) and estimate the rating of each aspect.

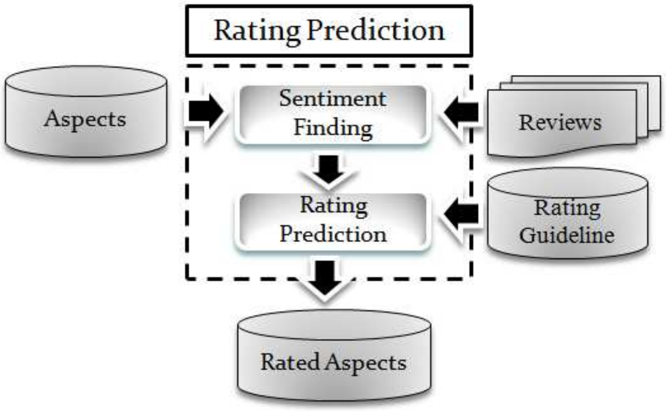


Figure 4.2: Rating prediction

For each aspect a_{im}, Opinion Digger first extracts the nearest sentiment to each occurrence of each candidate aspect $c_{ij} \in Group(a_{im})$ in the set of reviews of the item. Sentiments are usually the nearest adjectives in the same sentence segment which describe the quality of the aspect. Then the rating guideline of Epinions.com is used, and a k nearest neighbor (KNN) algorithm is applied to estimate the rating of each extracted sentiment. Wordnet [35] is used to compute similarity between adjectives for the KNN algorithm. As shown in Figure 4.3, in a 5-level orientation scale, most adjectives have two nearest neighbors, like defective which is placed between poor and terrible. Some of the adjectives, like those semantically placed above excellent or below poor have just one nearest neighbor. Therefore, we set k equal to 2 and use a 2-NN algorithm for aspect rating.

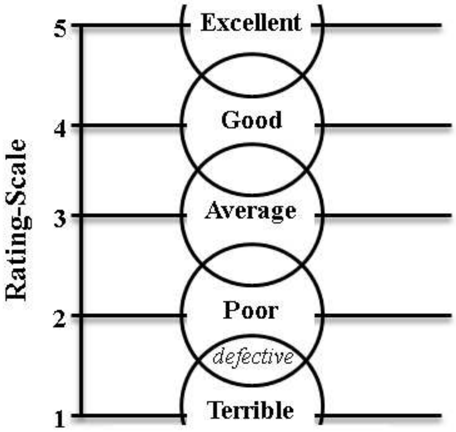


Figure 4.3: Sentiment rating space

For each sentiment snt Opinion Digger performs Breadth First Search (BFS) in the Wordnet synonymy graph with the maximum depth 5 to find two rated synonyms from the rating guideline. Then it uses a distance-weighted nearest neighbor algorithm with a continues-valued target function to return the weighted average of the ratings of 2-nearest neighbors as the estimated rating for the sentiment. So, the rating of the sentiment snt is equal to:

$$r_{snt} = \frac{\sum_i w_i \times r_i}{\Sigma_i w_i} \tag{4.4}$$

where w_i and r_i are the weight and rating of the neighbor n_i. w_i is defined based on the minimum path distance between neighbor n_i and sentiment snt in Wordnet hierarchy.

$$w_i = 1/distance(snt, n_i) \tag{4.5}$$

Weighted-distance methods give greater weights to closer neighbors and are more robust to noisy data. Figure 4.4 shows how our system estimates the rating of defective.

Finally, the system aggregates the ratings of all sentiments expressed about the candidates of an aspect to estimate its rating. For each item, Opinion Digger outputs a set of representative aspects and their estimated ratings.

$$w(terrible) = \frac{1}{distance(defective, terrible)} = \frac{1}{4}$$

$$w(poor) = \frac{1}{distance(defective, poor)} = 1$$

$$r(poor) = 2 \quad , \quad r(terrible) = 1$$

$$r(defective) = \frac{\Sigma_i w_i \times r_i}{\Sigma_i w_i} = \frac{1 \times 2 + 1/4 \times 1}{1/4 + 1} = 1.8$$

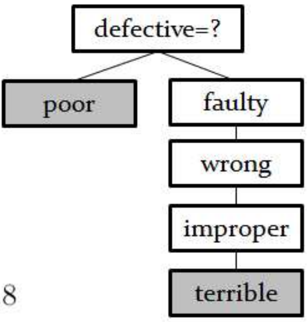


Figure 4.4: Rating prediction using Wordnet

4.4 Experimental Results

We evaluate our method from different points of view: the accuracy of opinion patterns, of aspect extraction, of aspect grouping, and of aspect rating. In the next subsection, we first briefly explain the dataset we use in our experiments.

4.4.1 Dataset

To evaluate the accuracy of Opinion Digger in extracting aspects and predicting their ratings, we need a labeled dataset. A labeled data set that has been used in some of the previous works is from Amazon.com [48]. Unfortunately none of the supervision data (know aspects and their ratings, rating guideline, and overall ratings of reviews) is available in this dataset. The main claim of the Opinion Digger is that using this available supervision data will improve the accuracy of aspect extraction and rating prediction.

To this end, we built a crawler to extract reviews and supervision data from the Epinions.com website. Our dataset contains 2.5K reviews in five product categories: camcorder, cellular phone, digital camera, DVD player, and MP3 player. We selected eight products in each category with different overall ratings. For each review we recorded the following information: known aspects and their ratings, full text review, and the overall rating of that review. Table 4.5 shows the distribution of reviews across the rating scale. For each category the number of reviews and the percentages of reviews that have a given overall rating are shown.

Table 4.6 also shows the distribution of known aspect ratings across the rating scale. As it is shown, the most frequent rating is 4, since it has been assigned to 38% of the known aspects. We

Table 4.5: Distribution of reviews across the rating scale

Category	#Rev.	1-star	2-star	3-star	4-star	5-star
Camcorder	197	33%	7%	8%	24%	28%
Cellular Phone	630	19%	9%	9%	27%	37%
Digital Camera	707	17%	6%	8%	26%	43%
DVD Player	324	24%	7%	10%	31%	28%
MP3 Player	625	12%	6%	8%	31%	43%
Overall	2483	21%	7%	8%	28%	36%

will use this rating in evaluation of aspect rating.

Table 4.6: Distribution of known aspects across the rating scale

Category	1-star	2-star	3-star	4-star	5-star
Camcorder	7.9%	6.9%	0.3%	39%	19.9%
Cellular Phone	4%	7.1%	0.2%	37%	33%
Digital Camera	3%	3.9%	0.1%	37.2%	44.3%
DVD Player	7.3%	9%	0.2%	37.3%	28.3%
MP3 Player	9.8%	9.7%	0.2%	41.9%	15.8%
Overall	5.4%	6.5%	0.2%	38%	33.4%

Regarding evaluation, we manually created a set of “true” aspects for each item as *gold standard*. We asked some judges to read the reviews for each item and provide a set of aspects for each based on reviews. We use the gold standard in the evaluation of aspect extraction.

4.4.2 Evaluation of Opinion Patterns

To evaluate the power of each opinion pattern we compute the percentage of extracted aspects and also non-aspects that match a specific pattern at least once in the review collection. In Table 4.7 the higher the percentage of aspects matching a pattern (first column) the higher the recall of that patterns for extracting actual aspects. Also, the higher the percentage of non-aspects matching a pattern (second column), the lower the precision of that pattern for extracting actual aspects.

Comparing the percentage of aspects and non-aspects matching each pattern shows that some of the patterns, like pattern 1, 2, and 3, have high recall and low precision on extracting aspects. In

Table 4.7: Percentage of extracted aspects and non-aspects matching each pattern

No.	Mined Patterns	% Aspects	% Non-Aspects
1	_JJ_ASP	100%	89%
2	_ASP_VB_JJ	98%	57%
3	_ASP_IN_JJ	97%	42%
4	_ASP_IN_NP_VB_JJ	68%	9%
5	_JJ_NP_IN_ASP	84%	32%
6	_JJ_IN_ASP	76%	21%
7	_JJ_NP_VB_ASP	52%	11%
8	_JJ_ASP_CC_ASP	62%	9%
9	_ASP_VB_VB_JJ	44%	7%

other words, while almost all of the extracted aspects match them, lots of non-aspects match too. On the other hand, the remaining patterns, especially patterns 4, 8, and 9, have high precision with lower recall. It means that the percentage of non-aspects matching them is very low. Using all of the patterns enables Opinion Digger to filter out non-aspects as much as possible.

4.4.3 Evaluation of Candidate Grouping

In this section we evaluate the performance of our candidate grouping method from two perspectives: quality of the aspect groups and quality of the selected representative for each group.

Our aspect grouping method is unsupervised and as comparison partners we select two state-of-the-art methods: an unsupervised method [93] and a minimally supervised method [121]. In [93] the authors proposed to employ WordNet to check if any synonym groups/sets exist among the candidates. They choose the top two frequent senses of a given word for finding its synonyms. That is, word A and word B will be grouped together only if there is a synset (synonym list) in WordNet containing A and B that appears in the top two senses of both words (we will refer to this method as TopSens). In [121], a Minimally Supervised Learning method (MSL) for grouping candidates is presented. This method starts from a small amount of seed words as supervised data for each group, and uses a bootstrapping mechanism to classify new candidates. A candidate c will be assigned to group a_j , if the context around c in the reviews is similar to the contexts of a_js members (verb, adjective, and noun phrases are used as contexts). We use the gold standard aspects as seed sets for this method. In the following we refer to this method as MSL. The performance of aspect grouping for different partners is evaluated using the Rand Index [139], a standard measure of clustering similarity often used to compare clusterings against a gold standard. The Rand Index measure is defined as follows:

$$RandIndex(P_i, P_m) = \frac{2(x+y)}{n \times (n-1)} \tag{4.6}$$

where P_i and P_m represent the grouping (partitions) produced by an algorithm i and manual labeling, respectively. The agreement of P_i and P_m is checked on their $n \times (n-1)$ pairs of candidates, where n is the number of extracted candidate. For each two candidates, P_i and P_m either assign them to the same group or to different groups. In Equation (4.6), x is the number of pairs belonging to the same group in both partitions, and y is the number of pairs belonging to different groups in both partitions.

Table 4.8: Rand Index of candidate grouping methods

Category	TopSens	MSL	OPD
Camcorder	0.75	0.85	0.91
Cellular Phone	0.65	0.78	0.84
Digital Camera	0.72	0.8	0.87
DVD Player	0.81	0.91	0.93
MP3 Player	0.67	0.78	0.83
Average	0.72	0.82	0.88

Not surprisingly, Table 4.8 shows that the Rand Index of MLS, which is a minimally supervised method, is higher than that of TopSens, which is unsupervised. It is also shown that, our unsupervised method (referred to as OPD) outperforms MLS. Although MLS showed good precision and recall for hotel reviews [121], it cannot achieve high Rank Index for product reviews. We believe the main reason is that MLS groups aspects based on their context. Since the contexts of different hotel aspects (like food, room, location, etc) are different from each other, MLS could perform well on hotel reviews. For example the context of the aspect food is {delicious, drink, good, etc.}, while the context of room is {large, clean, narrow, etc.} [121]. On the other hand, in product reviews, most of the aspects come in similar context and MLS cannot group them accurately. For example, the context of picture quality is {great, outstanding, like, etc.} and the context of sound quality is {outstanding, satisfying, great, etc.} which are very similar to each other. Note that, our proposed method for aspect grouping neither needs to have the number of groups as input (as it is needed in the clustering methods e.g. [42]), nor needs additional supervision data (as it is needed in classification methods e.g. [121]).

Another perspective for evaluating candidate grouping is the accuracy of selected aspects as representatives. Since none of the previous works chooses representative aspects for groups, there are no comparison partners. So, we evaluate this task by comparing selected representative aspects to the gold standard. Let C_i, A_i, and G_i are the sets of extracted candidate, representative aspects, and gold standard aspects for item i, respectively. The accuracy of candidate grouping for item i is equal to the percentage of gold standard aspects extracted ($C_i \bigcap G_i$) that are chosen as representatives of their aspect groups ($A_i \bigcap G_i$). Lu in [101] also used the same technique to evaluate the accuracy of aspect clusters' labels. The aspect grouping accuracy of item i is defined as follows:

$$GroupingAccuracy_i = \frac{|A_i \bigcap G_i|}{|C_i \bigcap G_i|} \tag{4.7}$$

As shown in Table 4.9, the average grouping accuracy over all items is 72%, meaning that gold standard aspects were mostly chosen as group representatives.

Table 4.9: Evaluation of candidate grouping

Item Category	Camcorder	CellPhone	DigitalCamera	DVDPlayer	MP3Player	Avg
Accuracy	76%	68%	72%	79%	64%	72%

4.4.4 Evaluation of Aspect Extraction

As is standard for evaluating frequency- and relation-based approaches, we evaluate the accuracy of aspect extraction using the provided set of true aspects. We compute precision and recall of aspect extraction versus this gold standard. Precision is equal to the number of extracted aspects which are true, over the total number of extracted aspects, and recall is equal to the percentage of true aspects which were extracted by the system (see details in Section 3.4).

As Opinion Digger is the first hybrid method proposed for the problem of aspect-based opinion mining, so there are no comparison partners in this category. However, there are several frequency-

and relation-based methods that can be compared with Opinion Digger to clarify the impact of combining the two ideas. As we explained in Section 3.2.1, frequency- and relation-based methods usually used various parameters that are manually or experimentally set for the given dataset. The details of these setting are normally not provided in the papers. However, the basic ideas are clear enough to implement a similar method. To this end, we compare our proposed aspect extraction technique with two methods: a frequency-based method similar to Feature-Based Summarization (FBS) [48] and a relation-based method similar to Opinion Observer [93].

The original FBS method first identifies frequent noun phrases as candidates and then applies two pruning techniques to remove those candidates whose words do not appear together in a specific order (meaningless) and those which are subsets of others (redundant). We implement a similar method by using the modified Apriori algorithm and applying the redundant pruning. The original Opinion Observer method applies an association rule mining algorithm on manually tagged reviews to find a set of POS patterns for extracting aspects in test data. We implement a similar method by using the mined POS patterns for extracting aspects. In other words, instead of using a set of manually tagged aspects, we used known aspects for learning opinion patterns.

Table 4.10: Average precision and recall of aspect extraction

Category	FBS		Opinion Observer		Opinion Digger	
	Precision	Recall	Precision	Recall	Precision	Recall
Camcorder	71%	65%	55%	78%	78%	62%
Cellular Phone	82%	74%	59%	84%	84%	68%
Digital Camera	78%	79%	57%	81%	79%	72%
DVD Player	69%	64%	53%	75%	76%	59%
MP3 Player	76%	72%	57%	79%	78%	66%
Average	75%	70%	56%	79%	79%	65%

Table 4.10 shows that the precision of FBS in extracting aspects is higher than that of Opinion Observer. The main reason for this higher accuracy is that the POS patterns used by Opinion Observer are match with a lot of words including non-aspects (Table 4.7 shows the percentages of non-aspects matching with each pattern). On the other hand, the recall of Opinion Observer is higher than FBS. The frequency threshold used in FBS filters out all infrequent aspects while Opinion Observer can identify these aspects. It might be interesting to mention that comparison of these

methods on short sentence segments presented in [93] shows the better performance of Opinion Observer as the grammatical structures of short sentence segments are simple and the learned patterns are more accurate.

Finally, comparing the precision of Opinion Digger with the comparison partners indicates the effectiveness of combining frequency- and relation-based methods. On the other hand, the recall of Opinion Digger is lower than both the other methods as it applies more restrictions for identifying aspects. Based on these experimental results we can conclude that selecting an accurate method for extracting aspects depends on the type of the given dataset, the available supervision data, and the relative importance of the precision and recall.

4.4.5 Evaluation of Rating Prediction

As our data set does not have the gold standard ratings for aspects, we evaluate the accuracy of rating prediction using two techniques: evaluating the accuracy of estimated ratings for known aspects, and evaluating the accuracy of the estimated overall rating for each review. Estimated ratings are evaluated using Ranking Loss which measures the average distance between the true and predicted numerical ratings (see details in Section 3.4).

In the following we compare the ranking loss of Opinion Digger with two comparison partners, including the simple Majority baseline used in [152] and the Polarity technique used in [48, 166, 196]. As mentioned in Section 4.4.1 the rating of 4 is the most common rating for all aspects and thus a prediction of all 4's for known aspects gives a Majority baseline and a natural indication of task difficulty. In the Polarity baseline, the rating of an aspect is determined by aggregating the polarity of the corresponding adjectives. This method starts from a set of seed adjectives which are labeled as positive or negative, and uses a bootstrapping mechanism to determine the polarity of a new adjective. A new adjective will be labeled as positive/negative, if it appears in the synset (synonym list in WordNet) of one of the positive/negative seed sets. In this work, we used the rated adjectives provided by the rating guideline and their synonyms from Wordnet as seed adjectives. We labeled adjectives with the rating of 1-2 as negative and 4-5 as positive.

Table 4.11 presents the ranking loss of the estimated ratings for known aspects in each category. It is shown that the Polarity technique performs better than the Majority baseline. In addition, the ranking loss of Opinion Digger is lower than that of Polarity technique. The main reason for this better performance is that our method looks for rated synonym up to 5 steps in synonymy graph. However, the Polarity technique can only use a fixed set of labeled adjectives.

Table 4.11: Ranking loss of estimated ratings for known aspects

Category	Majority	Polarity	Opinion Digger
Camcorder	1.06	0.88	0.56
Cellular Phone	1.05	0.84	0.52
Digital Camera	1.04	0.81	0.49
DVD Player	0.92	0.74	0.39
MP3 Player	1.07	0.91	0.6
Average	1.03	0.84	0.57

One of the limitation of evaluation using known aspects is that not all of the known aspects actually appear in the reviews so that they cannot be rated by our method. Therefore, we propose another technique for evaluating the accuracy of estimated aspect ratings based on the overall rating of products. This technique first estimates the overall rating of each product using extracted aspects and their estimated ratings, assuming that the overall rating of an item can be obtained by taking the average of the aspect ratings for that item. Then it compares the estimated overall rating with the actual overall rating expressed by reviewers.

Table 4.12 presents the ranking loss of estimated overall ratings for each category. These results are quite similar to those from Table 4.11. The average ranking loss of our method is 0.51 (in 5-star scale) which again demonstrates the high accuracy of the estimated aspect ratings. Comparing Tables 4.11 and 4.12, we observe that overall rating estimates are better when aspect rating estimates are better (e.g DVD player category).

Table 4.12: Ranking loss of estimated overall ratings

Category	Majority	Polarity	Opinion Digger
Camcorder	1.05	0.84	0.47
Cellular Phone	1.08	0.9	0.58
Digital Camera	1.09	0.92	0.63
DVD Player	0.96	0.79	0.42
MP3 Player	1.05	0.82	0.46
Average	1.05	0.85	0.51

4.5 Conclusion

In this work we proposed a hybrid approach, called Opinion Digger, for mining online reviews which provides a set of aspects and estimates their ratings. As input, Opinion Digger takes a set of known aspects and a rating guideline in addition to the review text for each item. It collects frequent noun phrases and then using known aspects it mines some opinion patterns from reviews to filter out non-aspects. It uses a novel method for both grouping synonymous candidate and selecting a good aspect representative for each group. While current works just determine whether people's opinion about a aspect is positive or negative, Opinion Digger precisely determines the strength of positiveness or negativeness of an opinion by estimating a rating in range [1,5] (as used in most of review Websites). The rating of an aspect is estimated based on the sentiments reviewers expressed about them and the rating guideline provided by the reviewing Website.

Evaluation of results supported our claim that using relationships between aspects can effectively improve the accuracy of aspect extraction. Opinion Digger outperformed all of the comparison partners in terms of precision of aspect extraction. However, its recall is lower than that of those methods because of applying more restrictions for extracting aspects. We conclude that selecting an accurate method for extracting aspects depends on the type of the given dataset, the available supervision data, and the relative importance of precision and recall. Opinion Digger also achieved high accuracy in grouping candidates and selecting representatives aspects. Finally, evaluation of rating estimation demonstrated the high accuracy of the estimated aspect ratings.

In this work we used exact matching for mining opinion patterns. However, this method fails to handle similar syntactic structures and therefore cannot be generalized for unseen data. Using methods based on tree kernels is a promising research direction that can address this limitation. Another important area for future work is the consideration of implicit aspects, e.g., ‘weight’ in the sentence “This camera is light”.

As discussed before, the need for manual tuning of various parameters makes the frequency- and relation-based methods hard to port to another dataset. So, in the next chapter we move on to model-based approach and propose a probabilistic graphical model to overcome this limitation by automatically learning the model parameters from the data.

Chapter 5

ILDA: Interdependent LDA Model

As discussed before, during the last decade several methods have been proposed to detect aspects and estimate their ratings from online reviews. Most of the early works are frequency- and relation-based approaches. One of the main limitations of these methods is that they require the manual tuning of various parameters which makes them hard to port to another dataset. Model-based techniques overcome this limitation by automatically learning the model parameters from the data. While supervised learning techniques (e.g., HMM and CRF) overcome the weakness of the previous approaches, they still need manually labeled data for training the models. So, recently researchers take advantages of unsupervised learning techniques, especially topic modeling.

Different extensions of the basic topic models, especially LDA, have been proposed to address the problem of aspect-based opinion mining. However, all of these models perform aspect identification and rating prediction in separate steps leading to the accumulation of errors. For example, a separate rating prediction algorithm will rate the sentiment 'long' equally for the aspects 'battery life' and 'shutter lag', although 'long' expresses a positive opinion for 'battery life' but a negative opinion for 'shutter lag'. In addition, most of the current works use the bag-of-words representation of reviews. As shown in [163], representing a document as a mixture of latent topics which generate all words of the document, can mainly be used in document clustering, by finding an overall topic of each document. However, in aspect-based opinion mining, the goal is not to cluster reviews, but to identify aspects and their corresponding ratings. For example, a topic modeling method applied to a collection of digital camera reviews is likely to infer some overall topics for that collection, such as 'Sony digital camera', and 'reviews of Sony'. Though these are valid topics, they do not represent product aspects. An aspect-based opinion mining model, on the other hand, tries to infer product aspects, such as 'zoom' and 'battery life', from the same collection of reviews. A solution that has been proposed in [101] is to first preprocess the reviews (chunk them into opinion phrases containing of a head term and a modifier) and then learn models that generate only opinion phrases, not all the words of a review.

In this work, we propose three probabilistic graphical models to jointly identify aspects and predict their ratings from online reviews. The first model is an extension of the PLSI model proposed in [101], and the second model is extending the standard LDA to generate a rated aspect summary of reviews. We consider these two models as baselines. As our main contribution, we introduce the *Interdependent Latent Dirichlet Allocation (ILDA)* model. We argue that ILDA is most natural for our problem since the underlying probabilistic assumptions (interdependency between aspects and ratings) are appropriate for the problem domain. We conduct experiments on a real life dataset from Epinions.com, demonstrating the improved effectiveness of the ILDA model in terms of the likelihood of a held-out test set, and the accuracy of aspects and aspect ratings. We also evaluate the performance of our model on real-life data sets from Amazon.com and TripAdvisor.com (will be discussed in the Chapter 7).

The remainder of the chapter is organized as follows. The next section discusses our contributions. Section 5.2 presents three probabilistic graphical models for the considered problem. Section 5.3 describes the inference algorithms for our proposed models. In Section 5.4 we report the results of our experimental evaluation. Finally, section 5.5 concludes the chapter with a summary and the discussion of future work.

5.1 Our Contributions

As discussed in the introduction, our goal is to provide a method to identify aspects and predict their ratings from online reviews without any human supervision. We use probabilistic graphical models, which represent each review as a mixture of latent aspects and ratings. We first extend the proposed model in [101] which is based on Probabilistic Latent Semantic Indexing (PLSI) model [47]. Then we extend the most well-known method for unsupervised modeling of documents, Latent Dirichlet Allocation (LDA) [14] to solve the considered problem. We make simple adaptations on both models for the aspect identification and rating prediction problem and consider them as baseline models. As our main contribution, we propose a novel model, called ILDA, for jointly extracting major aspects of items and predicting their ratings from online reviews. Unlike the previous topic modeling approaches that treat aspect identification and rating prediction as separate tasks, ILDA performs both tasks simultaneously in an unsupervised manner. We argue that considering the interdependency between aspects and ratings improves the performance of the model. To illustrate the importance of this interdependency, consider the following examples: 'low LCD resolution' and 'low price'. The sentiment 'low' expresses a negative opinion for 'LCD resolution', while it can be a positive opinion for 'price'. Treating rating prediction as a separate task, both aspects receive equal ratings. We also present algorithms for approximate inference and parameter estimation for the proposed LDA and ILDA models. We have conducted experiments on a real life dataset from Epinions.com. The experimental results show that ILDA consistently outperforms the baseline PLSI and LDA models. We also evaluate the performance of our model on real-life data sets from Amazon.com and TripAdvisor.com (see Chapter 7).

5.2 Probabilistic Graphical Models

In this section, we first describe our two baseline probabilistic models of reviews, noting their strengths and limitations: A PLSI model, and a multinomial LDA model. Then we introduce a novel model, ILDA, which models the interdependency between aspects and ratings. All of these models assume that aspects and their ratings can be represented by multinomial distributions and try to cluster head terms into aspects and modifiers into ratings.

5.2.1 PLSI Model

Probabilistic Latent Semantic Indexing (PLSI) [47] has recently been applied to many text mining problems. Lu et al. [101] applied the PLSI model on opinion phrases to identify aspects from reviews. However, as we described in section 3.2.2, they used PLSI only for aspect identification, and their model does not generate ratings for the identified aspects. We extend their PLSI model to identify aspects and predict their ratings simultaneously, as shown in Figure 5.1. Following the standard graphical model formalism, nodes represent random variables and edges indicate possible dependence. A random variable is a variable that can take on a set of possible different values, each with an associated probability. Shaded nodes are observed random variables and unshaded nodes are latent random variables. Finally, a box around groups of random variables is a 'plate' which denotes replication. The outer plate represents reviews and the inner plate represents opinion phrases. D and N are the number of reviews for the given item and the number of opinion phrases in each review, respectively. Since N is independent of all the other data generating variables (a and r), its randomness is generally ignored [14].

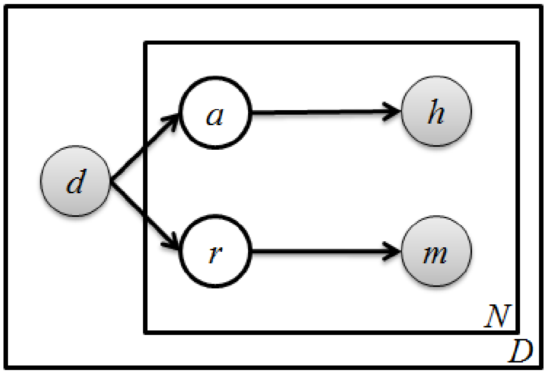


Figure 5.1: The PLSI model of reviews

To extend the PLSI model in [101] for our problem, we add the second row (dependency of the observed modifier m to the latent rating r, and the latent rating r to the observed review d). For each item in the data set, a PLSI model is generated to associate unobserved aspect a_n and rating r_n with each observation, i.e., with each opinion phrase $< h_n, m_n >$ in a review $d \in D$. The adapted generative PLSI model can be defined in the following way:

1. Select a review d from D with probability $P(d)$.
2. For each opinion phrase $< h_n, m_n >$, $n \in \{1, 2, ..., N\}$
   (a) Sample $a_n \sim P(a_n|d)$ and $r_n \sim P(r_n|d)$.
   (b) Sample $h_n \sim P(h_n|a_n)$ and $m_n \sim P(m_n|r_n)$.

Translating this process into a joint probability distribution results in the expression:

$$P(d, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}) = P(d) \prod_{n=1}^{N} [P(a_n|d)P(r_n|d)P(h_n|a_n)P(m_n|r_n)] \tag{5.1}$$

An equivalent symmetric version of the model can be obtained by inverting the conditional probabilities $P(a_n|d) = P(d|a_n)P(a_n)$ and $P(r_n|d) = P(d|r_n)P(r_n)$ with the help of Bayes' rule. Adopting the likelihood principle, $P(d)$ and $P(\boldsymbol{a}, \boldsymbol{r}|d)$ can be determined by maximization of the log-likelihood. The standard procedure for maximum likelihood estimation in latent variable models is the Expectation Maximization (EM) algorithm [28]. EM alternates two steps: expectation (E-step) which computes the posterior probabilities for latent variables, and maximization (M-step) which updates the parameters. The E-step equation for the PLSI model is:

$$P(a_n, r_n|d, h_n, m_n) = \frac{P(d|a_n)P(a_n)P(d|r_n)P(r_n)P(h_n|a_n)P(m_n|r_n)}{\sum_{a,r} P(d|a)P(a)P(d|r)P(r)P(h_n|a)P(m_n|r)} \tag{5.2}$$

and the M-step formulas are:

$$P(d|a_n) \propto \sum_{r,h,m} P(a_n, r|d, h, m)n(d, h, m) \tag{5.3}$$

$$P(a_n) \propto \sum_{d,r,h,m} P(a_n, r|d, h, m)n(d, h, m) \tag{5.4}$$

$$P(d|r_n) \propto \sum_{a,h,m} P(a, r_n|d, h, m)n(d, h, m) \tag{5.5}$$

$$P(r_n) \propto \sum_{d,a,h,m} P(a, r_n|d, h, m)n(d, h, m) \tag{5.6}$$

$$P(h_n|a_n) \propto \sum_{d,r,m} P(a_n, r|d, h_n, m)n(d, h_n, m) \tag{5.7}$$

$$P(m_n|r_n) \propto \sum_{d,a,h} P(a, r_n|d, h, m_n)n(d, h, m_n) \tag{5.8}$$

where $n(d, h, m)$ denotes the frequency of the phrase $< h, m >$ occurring in review d. The EM algorithm obtains a local maximum of the log-likelihood by alternating E-step (5.2) with M-steps (5.3)-(5.8). It is important to note that d is a multinomial random variable with as many possible values as there are training reviews (i.e a dummy index into the list of reviews in the training set) and the PLSI model learns the $P(\boldsymbol{a}, \boldsymbol{r}|d, \boldsymbol{h}, \boldsymbol{m})$ only for those reviews on which it is trained. For this reason, PLSI is not a well-defined generative model of reviews [14]. Furthermore, the number of PLSI parameters which must be estimated grows linearly with the number of training reviews which causes overfitting. One reasonable approach to avoid overfitting is assigning probability to previously unseen data by marginalizing over seen data [14]. We use this approach to smooth the parameters of the PLSI model for acceptable predictive performance as follows:

$$P(\boldsymbol{h}, \boldsymbol{m}) = \sum_{d} \prod_{n=1}^{N} \sum_{a,r} P(d|a, r)p(a, r)P(h_n|a)P(m_n|r) \tag{5.9}$$

5.2.2 Multinomial LDA

The Latent Dirichlet Allocation (LDA) model is a generative probabilistic model for collections of discrete data such as text corpora [14]. The basic idea is that each item of a collection is modeled as a finite mixture over an underlying set of latent variables.

The aspect identification and rating prediction problem can be modeled as an extension of LDA (Figure 5.2). We make a simple adaption to the basic LDA by adding the second row (dependency among m, r, and θ) to the basic LDA model. This model is similar to the GM-LDA model presented in [12] for the annotation of images. In our adapted LDA model a review is assumed to be generated by first choosing a value of θ, and then repeatedly sampling N opinion phrases $< h_n, m_n >$ conditional on the chosen value of θ. Similar to [12], where θ represents the image/caption pairs, in our multinomial LDA model we can view θ as a high-level representation of the collection of aspect/rating pairs. For every pair, θ contains the probability of generating that combination of aspect and rating. The variable θ is sampled once per review, and is held fixed during the process of generating opinion phrases for that review. After sampling θ, the latent variables a_n and r_n are sampled independently (conditional independency) and then an opinion phrase $< h_n, m_n >$ is sampled conditional on the sampled a_n and r_n. Our adapted LDA model assumes the following generative process:

1. Sample $\theta \sim Dir(\alpha)$.
2. For each opinion phrase $< h_n, m_n >$, $n \in \{1, 2, ..., N\}$
   (a) Sample $a_n \sim P(a_n|\theta)$ and $r_n \sim P(r_n|\theta)$
   (b) Sample $h_n \sim P(h_n|a_n, \beta)$ and sample $m_n \sim P(m_n|r_n, \pi)$

$P(h_n|a_n, \beta)$ and $P(m_n|r_n, \pi)$ are multinomial distributions conditioned on aspect a_n and rating r_n, respectively. The resulting joint distribution is as follows:

$$P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \theta|\alpha, \beta, \pi) = P(\theta|\alpha) \prod_{n=1}^{N} [P(a_n|\theta)P(r_n|\theta)P(h_n|a_n, \beta)P(m_n|r_n, \pi)] \tag{5.10}$$

The key inferential problem is to compute the posterior distribution of the latent variables given a review $< \boldsymbol{h}, \boldsymbol{m} >$ (collection of N opinion phrases $< h_n, m_n >$):

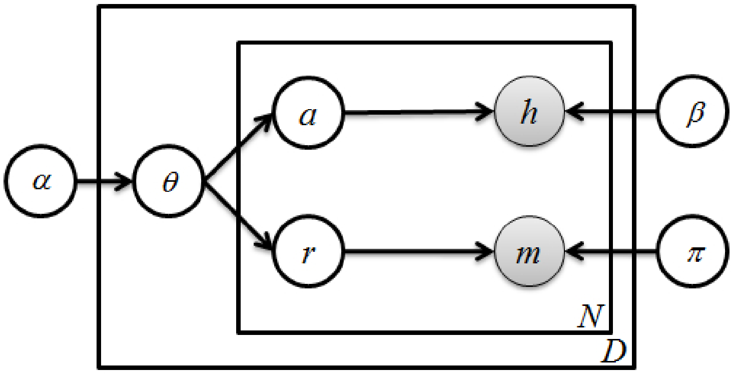


Figure 5.2: The Multinomial LDA (MLDA) model of reviews

$$P(\boldsymbol{a}, \boldsymbol{r}, \theta | \boldsymbol{m}, \boldsymbol{h}, \alpha, \beta, \pi) = \frac{P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \theta | \alpha, \beta, \pi)}{P(\boldsymbol{h}, \boldsymbol{m} | \alpha, \beta, \pi)} \tag{5.11}$$

Similar to the basic LDA model, due to the coupling between θ with β and π the conditional distribution of latent variables given observed data is intractable to compute. Although the posterior distribution is intractable for exact inference, a wide variety of approximate inference algorithms can be considered for LDA models [14]. In this work, we use variational inference to compute an approximation for the posterior distribution.

The multinomial LDA model overcomes both of the PLSI problems by using a latent random variable θ rather than a large set of individual parameters which are explicitly linked to the training reviews [14]. However, conditional on the latent variable θ, the multinomial LDA model generates aspects and ratings independently, and so the dependency between specific aspects and specific ratings is ignored. We will show experimentally that due to the lack of this dependency, the multinomial LDA model cannot capture the correspondence between aspects and ratings.

5.2.3 Interdependent LDA

In this section, we introduce Interdependent Latent Dirichlet Allocation (ILDA) which models the conditional interdependency between the latent aspects and ratings. As pointed out already, one and the same sentiment word may show different opinions for different aspects, and methods modeling aspects and ratings separately miss this interdependency between aspects and expressed sentiments.

We present the ILDA model, shown in Figure 5.3, to overcome this weakness by jointly modeling latent aspects and ratings. ILDA can be viewed as a generative process that first generates

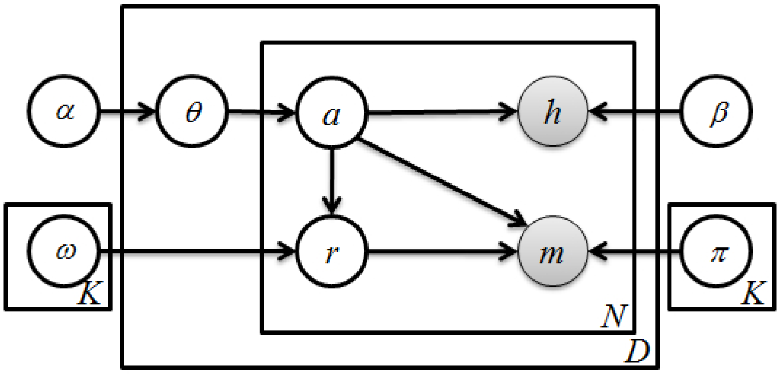


Figure 5.3: The ILDA model of reviews

an aspect and subsequently generates its rating. In particular, for generating each opinion phrase, ILDA first generates an aspect a_n from an LDA model. Then it generates a rating r_n conditioned on the sampled aspect a_n. Finally, a head term h_n is drawn conditioned on a_n and a modifier m_n is sampled conditional on both a_n and r_n. Formally, the k-factor ILDA model assumes the following generative process for a review (collection of opinion phrases):

1. Sample $\theta \sim Dir(\alpha)$.
2. For each opinion phrase $< h_n, m_n >$, $n \in \{1, 2, ..., N\}$
   (a) Sample $a_n \sim Mult(\theta)$
   (b) Sample $r_n \sim P(r_n|a_n, \omega)$, a multinomial distribution conditioned on the aspect a_n.
   (c) Sample $h_n \sim P(h_n|a_n, \beta)$ and sample $m_n \sim P(m_n|a_n, r_n, \pi)$

where $P(h_n|a_n, \beta)$ and $P(m_n|a_n, r_n, \pi)$ are multinomial distributions. ILDA thus specifies the following joint distribution:

$$P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \theta|\alpha, \boldsymbol{\omega}, \beta, \boldsymbol{\pi}) = \\ P(\theta|\alpha)\prod_{n=1}^{N}[P(a_n|\theta)P(r_n|a_n, \boldsymbol{\omega})P(h_n|a_n, \beta)P(m_n|a_n, r_n, \boldsymbol{\pi})] \quad (5.12)$$

The dependency assumption of the ILDA model overcomes the lack of correspondence in the multinomial LDA model, where the head terms and modifiers are generated independently, conditional on the latent variable θ. In the ILDA model, the head term is conditional on a generated aspect

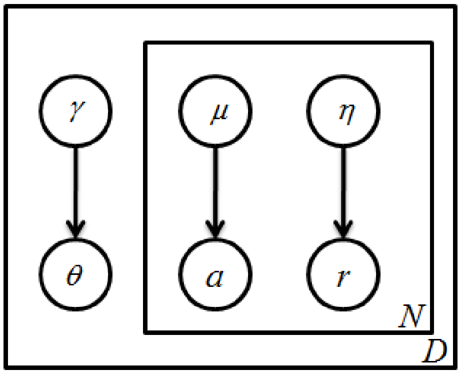


Figure 5.4: Graphical model representation of the variational distribution used to approximate the posterior in MLDA and ILDA models

and the modifier must be conditional on the selected aspect and a rating which is correspondent to that aspect. In fact, the ILDA model captures the phenomenon that the head term is generated first and then the modifier rates the head term.

5.3 Inference and Estimation

In this section, we describe approximate inference and parameter estimation for the multinomial LDA and ILDA models, adopting a variational method.

5.3.1 Variational Inference

Computing the posterior distribution of the latent variables for both multinomial LDA and ILDA models is intractable. A common way to obtain a tractable lower bound is to consider simple modifications of the original graphical model [14]. In particular, we simplify these models into the graphical model shown in Figure 5.4. This model specifies the following variational distribution on the latent variables:

$$Q(\boldsymbol{a}, \boldsymbol{r}, \theta | \boldsymbol{\mu}, \boldsymbol{\eta}, \gamma) = Q(\theta|\gamma) \prod_{n=1}^{N} [Q(a_n|\mu_n) Q(r_n|\eta_n)] \tag{5.13}$$

where the Dirichlet parameter γ and the multinomial parameters $(\mu_1, ..., \mu_n)$ and $(\eta_1, .., \eta_n)$ are free variational parameters.

To have a good approximation, the KL-divergence between the variational distribution and the

Algorithm 1 Variational Inference Algorithm for ILDA

```
initialize $\mu_{ni}^0 := 1/k$ for all $i$ and $n$
initialize $\eta_{nj}^0 := 1/5$ for all $j$ and $n$
initialize $\gamma_i^0 := \alpha_i + N/k$ for all $i$
repeat
  for $n = 1$ to $N$ do
    for $i = 1$ to $k$ do
      $\mu_{ni}^{t+1} := \beta_{ix} \prod_j^5 (\omega_{ij}\pi_{ijy})^{\eta_{nj}^t} \exp(\psi(\gamma_i^t))$
    end for
    normalize $\mu_{ni}^{t+1}$ to sum to 1
    for $j = 1$ to $5$ do
      $\eta_{nj}^{t+1} := \prod_i^K (\omega_{ij}\pi_{ijy})^{\mu_{ni}^t}$
    end for
    normalize $\eta_{nj}^{t+1}$ to sum to 1
  end for
  $\gamma^{t+1} := \alpha + \sum_{n=1}^N \mu_n^{t+1}$
until convergence
```

true posterior needs to be minimized. This minimization can be achieved via an iterative method. Taking derivatives of the KL-divergence with respect to variational parameters and setting them equal to zero, we obtain the update equations. The pseudo-code of the variational inference procedure for the ILDA model is presented in Algorithm 1. These update equations are invoked repeatedly until the change in KL-divergence is small.

In Algorithm 1, β_{ix} is $P(h_n^x = 1|a^i = 1)$ for the appropriate x. Recall that each h_n is a vector with exactly one component equal to one; we can select the unique x such that $h_n^x = 1$. In the same way π_{ijy} is $P(m_n^y = 1|a^i = 1, r^j = 1)$ for the appropriate y. With the approximate posterior in hand, we can find a lower bound on the joint probability, $P(\boldsymbol{a}, \boldsymbol{r}, \theta)$. In the next section we use this lower bound to estimate the ILDA parameters.

The variational inference update formulas for the Multinomial LDA model are as follows[1]:

$$\mu_{ni}^{t+1} := \beta_{ix} \prod_{j=1}^{5} \exp(\eta_{nj}^t \psi(\gamma_{ij}^t)) \tag{5.14}$$

$$\eta_{ni}^{t+1} := \pi_{jy} \prod_{i=1}^{K} \exp(\mu_{ni}^t \psi(\gamma_{ij}^t)) \tag{5.15}$$

[1] The detailed derivation of the variational EM algorithm for both LDA and ILDA models is available at `http://www.sfu.ca/~sam39/ILDA`

$$\gamma^{t+1} := \alpha + \sum_{n=1}^{N} \mu_n^{t+1} \eta_n^{t+1} \tag{5.16}$$

5.3.2 Parameter Estimation

Given a corpus of reviews $D = \{d_1, d_2, ..., d_D\}$ about a specific item, we want to find model parameters that maximize the (marginal) log likelihood of the data:

$$\ell(\alpha, \beta, \pi, \omega) = \sum_{d=1}^{D} \log P(\boldsymbol{h_d}, \boldsymbol{m_d} | \alpha, \beta, \boldsymbol{\pi}, \boldsymbol{\omega}) \tag{5.17}$$

As we have described, the computation of the posterior distribution, $P(\boldsymbol{h}, \boldsymbol{m} | \alpha, \beta, \boldsymbol{\pi}, \boldsymbol{\omega})$, is intractable and therefore we use variational inference to obtain a tractable lower bound on the log likelihood. We can thus find approximate estimates for the ILDA model via an alternative *Variational EM* procedure [14]. The variational EM maximizes the lower bound of the log likelihood with respect to the variational parameters, and then for fixed values of the variational parameters, maximizes the lower bound with respect to the model parameters.

To maximize with respect to each variational parameter, we take derivatives with respect to it and set it to zero. The derivation yields the following iterative algorithm:

1. (E-step) For each review, find the optimizing values of the variational parameters γ^*, μ^*, and η^* (using Algorithm 1).

2. (M-step) Maximize the resulting lower bound on the log likelihood with respect to the model parameters α, β, $\boldsymbol{\pi}$, and $\boldsymbol{\omega}$.

The variational EM algorithm alternates between these two steps until the bound on the expected log likelihood converges. The M-step updates for the conditional multinomial parameters β and $\boldsymbol{\pi}$ in the ILDA model are as follows:

$$\beta_{ix} = \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mu_{dni}^* h_{dn}^x \tag{5.18}$$

$$\pi_{ijy} = \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mu_{dni}^* \eta_{dnj}^* m_{dn}^y \tag{5.19}$$

$$\omega_{ij} = \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mu^*_{dni} \eta^*_{dnj} \tag{5.20}$$

The M-step update for the Dirichlet parameter α is implemented using an efficient Newton-Raphson method in which the Hessian is inverted in linear time [14]. The Newton-Raphson optimization technique finds a stationary point of a function by iterating:

$$\alpha_{new} = \alpha_{old} - H(\alpha_{old})^{-1} g(\alpha_{old}) \tag{5.21}$$

where $H(\alpha)$ and $g(\alpha)$ are the Hessian matrix and gradient respectively at the point α.

5.3.3 Smoothing

Overfitting has always been a serious problem when working with conditional distributions [14]. A new review is very likely to contain head terms or modifiers that did not appear in any of the reviews in a training corpus. Maximum likelihood estimate of the multinomial parameters β and $\boldsymbol{\pi}$ assign zero probability to such head terms or modifiers, and so zero probability to new reviews. The standard approach to dealing with this problem is smoothing the parameters which are dependent to the observed data, by assigning positive probability to all vocabulary words whether or not they are observed in the training set.

In this work we use Dirichlet smoothing [14] which places Dirichlet priors on the multinomial parameters β and $\boldsymbol{\pi}$ (which are dependent to the observed data) to avoid overfitting. The Dirichlet smoothing treats β and $\boldsymbol{\pi}$ as random matrices whose rows independently drawn from exchangeable Dirichlet distributions. An exchangeable Dirichlet distribution is simply a Dirichlet distribution with a single scalar parameter. In other words, we now consider $\beta_i \sim Dirichlet(\rho, \rho, ..., \rho)$ and $\pi_{ij} \sim Dirichlet(\varrho, \varrho, ..., \varrho)$ where ρ and ϱ are scalar parameters.

A variational approach can again be used to find an approximation to this posterior distribution. We adopt a variational approach that places a separable distribution on the random variables β, π, θ, a, and r:

$$Q(\boldsymbol{a}, \boldsymbol{r}, \theta, \beta, \boldsymbol{\pi} | \boldsymbol{\mu}, \boldsymbol{\eta}, \gamma, \phi, \boldsymbol{\varphi}) = \prod_{i=1}^{K} Dir(\beta_i|\phi_i)$$
$$\prod_{i=1}^{K} \prod_{j=1}^{5} Dir(\pi_{ij}|\varphi_{ij}) \prod_{d=1}^{D} Q_d(\boldsymbol{a}_d, \boldsymbol{r}_d, \theta_d | \boldsymbol{\mu}_d, \boldsymbol{\eta}_d, \gamma_d) \quad (5.22)$$

where ϕ and $\boldsymbol{\varphi}$ are variational Dirichlet parameters for β and $\boldsymbol{\pi}$, respectively, and $Q_d(\boldsymbol{a}_d, \boldsymbol{r}_d, \theta_d | \boldsymbol{\mu}_d, \boldsymbol{\eta}_d, \gamma_d)$ is the variational distribution defined in Equation (5.13). The only change to our M-step algorithm is to replace the maximization with respect to β and $\boldsymbol{\pi}$ with the following variational updates:

$$\phi_{ix} = \rho + \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mu^*_{dni} h^x_{dn} \quad (5.23)$$

$$\varphi_{ijy} = \varrho + \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mu^*_{dni} \eta^*_{dnj} m^y_{dn} \quad (5.24)$$

Bayesian methods often assume a noninformative prior which means setting $\rho = \varrho = 1$ [12]. Iterating these equations to convergence yields an approximate posterior distribution on β, $\boldsymbol{\pi}$, θ, $\boldsymbol{a}$, and $\boldsymbol{r}$.

5.3.4 Model Selection

To find the optimal number of aspects we compute the Bayesian Information Criterion (BIC) [148]. BIC is a criterion for model selection among a class of parametric models with different numbers of parameters. When estimating model parameters using maximum likelihood estimation, it is possible to increase the likelihood by adding parameters, which may however result in overfitting. The BIC resolves this problem by introducing a penalty term for the number of parameters in the model.

Let n be the number of opinion phrases in the given test set, k be the number of free parameters to be estimated (number of aspects), and $\mathcal{L}$ be the maximized value of the likelihood function for the estimated model. The formula for the BIC is:

$$BIC = -2\ln(\mathcal{L}) + k\ln(n) \quad (5.25)$$

Given any two learned models, the model with the lower value of BIC is the one to be preferred. Hence, lower BIC implies either fewer free parameters, better fit, or both. Figure 5.5 shows BIC

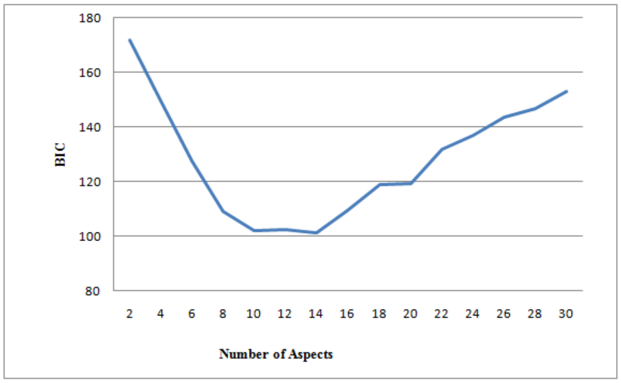


Figure 5.5: BIC of ILDA for different numbers of aspects (k)

values of the ILDA model for one representative product in the digital camera category. For this product, the BIC value of the model reaches its minimum for $k = 14$, i.e. ILDA identifies 14 aspect clusters in the given reviews. For each product, we train models for a range of values of k and pick the model with the optimal k, i.e. lowest BIC.

5.4 Experimental Results

In this section, we experimentally compare the three models proposed in this paper, i.e., PLSI, Multinomial LDA (MLDA), and ILDA. We also test a simple multinomial model (ML), that treats the head terms and modifiers as independent multinomials, as a simple baseline method. An extensive comparison of ILDA with the state-of-the-art LDA models will be presented in the next chapter.

As discussed in Section 3.4, a standard approach for evaluation of graphical models is comparing the achieved likelihoods of a held-out test set. So, we evaluate ILDA by comparing its obtained likelihood on a held-out test set with the likelihoods achieved by the comparison partners. However, the accuracy of aspect identification and rating prediction cannot be inferred easily by this evaluation. Therefore, to make the evaluation stronger we compute the accuracy of the model's results using a standard measure of clustering. While accuracy evaluation provides more information about the performance of the learned model, it needs a truly labeled test set which makes it subjective in contrast with the likelihood evaluation approach which is completely objective.

In the next subsections, we first briefly describe our dataset and then present the evaluation of the models in terms of test set likelihood and accuracy of aspect identification and rating prediction.

5.4.1 Dataset

In this work, we used the Epinions data set that we crawled in our previous work. We use all of the frequent noun phrases and their nearest adjectives as opinion phrases. In Table 6.2, for each category the number of reviews, opinion phrases, and phrases per product (on average) are shown.

Table 5.1: Statistics of the dataset

Category	#Reviews	#Opinion Phrases	#Phrases per Product
Camcorder	197	1,694	211.77
Cel. Phone	630	7,642	955.23
Dig. Camera	707	10,435	1304.41
DVD Player	324	3,707	463.32
Mp3 Player	625	6,131	766.41
Overall	2,483	29,609	3701.14

We asked some judges to label each opinion phrase with a pair of $< g_a, g_r >$ based on the given head term and corresponding modifier; where g_a is a number showing the aspect cluster of the given head term, and g_r is the rating of the given modifier with respect to the given head term (considering the correspondence between head term and modifier). This gold standard is used in the evaluation of aspect identification and rating prediction.

5.4.2 Test Set Likelihood

We held out 20% of the reviews for testing purposes and used the remaining 80% to learn the model. Our goal is to achieve high likelihood on a held-out test set. Note that, unlike in text modeling problems which learn one model from the whole collection of documents, in aspect-based opinion mining one independent model is learned per item. In the following, we present evaluation results for each category which is the average of the results of the products of that category.

To evaluate how well a model fits the data, we computed the perplexity of the held-out test set on all models for various values of aspects, k. The perplexity is monotonically decreasing in the likelihood of the test data, and a lower perplexity score indicates better performance (discussed in Section 3.4). Figure 5.6 shows the perplexity of different models for different product categories. As expected, the latent variable models perform better than the simple multinomial model (ML). The MLDA model which suffers from neither of the PLSI problems, consistently performs better.

Most notably, ILDA performs much better than either MLDA or PLSI and provides a better fit which indicates that ILDA models the reviews more accurately.

The major reason for significant performance enhancement of ILDA and MLDA compared to PLSI is that they effectively capture the latent semantic association among aspects. Moreover, ILDA consistently outperforms MLDA. We believe that this is due to the fact that ILDA captures the interdependency between latent aspects and the modifiers used to rate them. Also, it is notable that all the models perform better when the size of the training dataset is larger (e.g., digital camera category).

5.4.3 Accuracy of Aspect Clusters

In this section, we evaluate the accuracy of identifying the k major aspect clusters in the given test set. Since all of the proposed models are soft clustering techniques, for each head term the cluster with the highest probability is selected as its aspect cluster. For each model, the accuracy of identified aspects is evaluated using the Rand Index [139], a standard measure of clustering similarity often used to compare clusterings against a gold standard (discussed in Section 4.4.3).

Table 5.2 presents the Rand Index (the higher the better) of different models for the optimal numbers of aspects, i.e., for the k with the minimum BIC value for that model. The Rand Index of aspect identification for each category is the average of the Rand Index of its products. Not surprisingly, Table 5.2 shows that ILDA and MLDA achieve better accuracy than PLSI in all of the categories and ILDA clearly outperforms MLDA.

Table 5.2: Rand Index of aspect clusters for different models

Category	PLSI	MLDA	ILDA
Camcorder	0.59	0.68	0.8
Cellular Phone	0.64	0.76	0.86
Digital Camera	0.65	0.79	0.88
DVD Player	0.62	0.73	0.81
Mp3 Player	0.63	0.74	0.84
Average	0.62	0.74	0.83

5.4.4 Accuracy of Rating Clusters

For each model, the accuracy of rating clusters is also evaluated using the Rand Index. The Rand Index scores of different models are shown in Table 5.3. Note that the number of rating clusters for all models is known and fixed $(k = 5)$. Again, for each modifier the cluster with the highest probability is selected as its rating cluster.

Again, ILDA and MLDA achieve better accuracy than PLSI because of capturing the latent semantic association among ratings. Moreover, by capturing the interdependency between aspects, ratings and modifiers, ILDA outperforms MLDA. Note that in Tables 5.2 and 5.3, all the models perform better for digital camera category which has the largest training set.

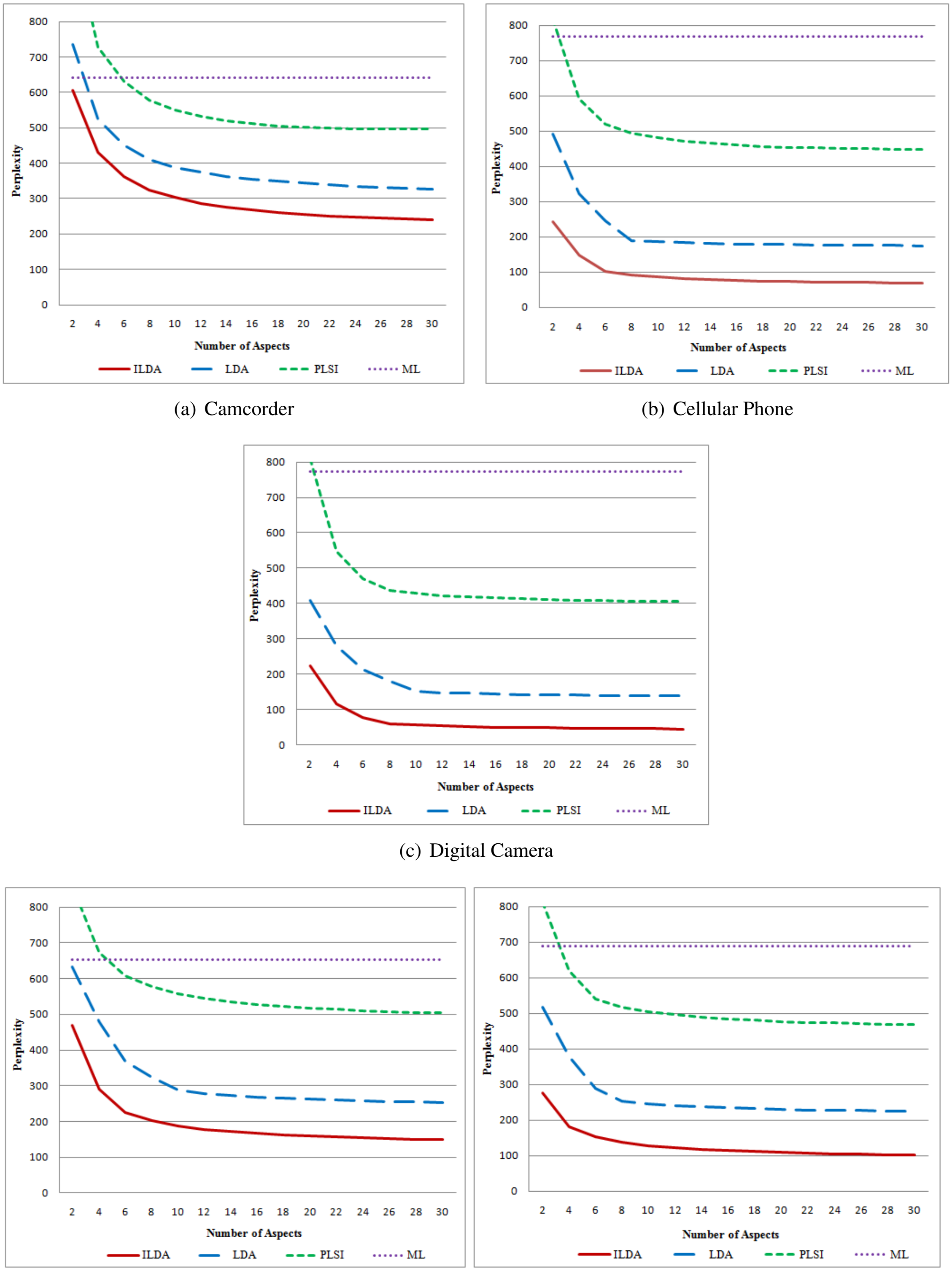
Table 5.3: Rand Index of rating clusters for different models

Category	PLSI	MLDA	ILDA
Camcorder	0.41	0.49	0.68
Cellular Phone	0.48	0.56	0.74
Digital Camera	0.5	0.58	0.76
DVD Player	0.44	0.53	0.71
Mp3 Player	0.45	0.56	0.74
Average	0.46	0.54	0.73

5.5 Conclusion

In this work we proposed a probabilistic graphical model based on LDA, called ILDA, that learns a set of aspects and corresponding ratings from a collection of reviews. The reviews are first preprocessed into a collection of opinion phrases and the model is then trained on a bag-of-phrases. We performed an experimental evaluation on a real life data set from the Epinions.com and compared ILDA against baseline PLSI and multinomial LDA models. ILDA clearly outperformed all of the comparison partners in terms of likelihood of the test set, and accuracy of aspect and rating clusters. We argue that the major reason for the consistent enhancement is that ILDA better captures the interdependency between latent aspects and ratings.

In the next chapter, we will compare the performance of ILDA with the state-of-the-art LDA models on a vary large data set. We will elaborate the ILDA's strengths and weaknesses in compare to these models and will discuss several directions for future research.



(a) Camcorder

(b) Cellular Phone

(c) Digital Camera

(d) DVD Player

(e) Mp3 Player

Figure 5.6: Perplexity results for PLSI, MLDA, and ILDA for different product categories

Chapter 6

On the Design of LDA Models

As discussed in Chapter 3, in the past few years several probabilistic graphical models have been proposed to address the problem of aspect-based opinion mining. Most of these models are based on LDA which is typically used in topic modeling. While these models have a lot in common, there are some characteristics that distinguish them from each other. In the following we point out some of the main distinctive features:

- Modeling words using one latent variable vs. having separate latent variables for aspects and ratings.
- Modeling all words of the reviews vs. modeling only opinion phrases.
- Modeling the dependency between aspects and ratings vs. modeling them independently.
- Using only review texts as input vs. also using additional input data, e.g., a review's overall rating.

These features correspond to major decisions that must be made in the design of an LDA model. While research papers typically claim that a new model outperforms the existing ones, it is normally not shown why and in which scenarios the proposed model performs better than another one. For example, let us consider a proposed model A generating all words which considers the dependency between aspects and their ratings and takes the overall rating of reviews as an additional observed variable. The comparison of model A with the basic LDA model on a dataset with a large number of reviews per item shows better performance of model A. The problem with this experimental evaluation is that it does not reveal whether the better performance of model A is due to the dependency assumption between aspects and ratings or due to the additional observed data. It is also not clear whether model A still performs better for another dataset containing items with few reviews.

We argue that the best choice for some design decision may depend on other design decisions and on the content and the size of the dataset. So, in this work we do not propose yet another LDA model for the problem of aspect-based opinion mining, but we present design guidelines for such models. To derive these guidelines, we discuss a series of increasingly sophisticated probabilistic graphical models based on LDA. We start with the basic LDA model and then gradually extend the model by adding latent and observed variables as well as dependencies. The discussed models are as follows:

- LDA: The basic LDA model proposed in [14] which learns general topics of reviews using all words of the training reviews.
- S-LDA: An extension of LDA where the model learns both aspects and ratings from reviews.
- D-LDA: Extension of S-LDA considering the dependency between aspects and their ratings.
- PLDA: The basic LDA model which learns general topics of reviews from opinion phrases.
- S-PLDA: An extension of PLDA where the model learns both aspects and ratings from phrases.
- D-PLDA: An LDA-based model learning aspects and their corresponding ratings from opinion phrases while considering the dependency between the aspects and ratings.

We argue that these six models represent the essence of the major published methods and allow us to tease apart the impact of various design decisions. For example, the comparison of S-LDA and S-PLDA against D-LDA and D-PLDA shows whether the dependency of aspects and ratings improves the performance, independent from the type of observed data (all words or opinion phrases).

Since there is no benchmark dataset for the problem of aspect-based opinion mining, current works have been evaluated on different data sets. However, one dataset may have items with only few reviews, while the items of another dataset may have been selected to have at least a few hundred reviews. We crawled the well-known reviewing website, Epinions.com, and built a very large dataset containing 505,978 reviews about 94,792 products from 257 different product categories. We made the dataset publicly available for research purposes[1]. We evaluate the performance of the six models in terms of the likelihood of a held-out test set. To measure the impact of the training set size, we

[1] `http://www.sfu.ca/~sam39/Datasets/EpinionsReviews/`

perform experiments for different subsets of products with different numbers of reviews. We also evaluate the accuracy of the models in aspect identification and rating prediction (precision, recall, and mean squared error) on a labeled dataset and find a strong correlation between model perplexity and accuracy.

As a novel technical contribution, we present a method for extracting opinion phrases based on grammatical relations provided by a dependency parser. This technique promises to generate opinion phrases more accurately than current methods that consider only syntactic properties such as the proximity of words.

The remainder of the chapter is organized as follows. In the next section we briefly discuss our contribution. Section 6.2 presents different LDA models for the considered problem. Section 6.3 describes the inference and estimation techniques for the presented models. The proposed technique for extracting opinion phrases is discussed in 6.4. In Section 6.5 we report the results of our experimental evaluation. Finally, Section 6.6 concludes the chapter with a summary of our design guidelines and the discussion of future work.

6.1 Our Contributions

As discussed in the introduction, our goal is to present a set of design guidelines for LDA models for learning aspects and their ratings from reviews. In this work, we focus on the following questions:

- Is it better to have separate latent variables for aspects and ratings?
- Is it better to assume dependency between ratings and aspects?
- Is it better to learn from bag-of-words or preprocess the reviews and learn from opinion phrases?
- Which preprocessing technique for extracting opinion phrases works best?
- Does the answer to the above questions differ for items with few reviews and items with many reviews?

We start our investigation with the basic LDA model [14] and then gradually extend the model by considering different probabilistic assumptions. In summary we discuss five LDA-based models with various underlying assumptions for our problem: *S-LDA* extends LDA by assuming the review is generated by a set of aspects and their ratings. *D-LDA* adds the dependency between aspects and

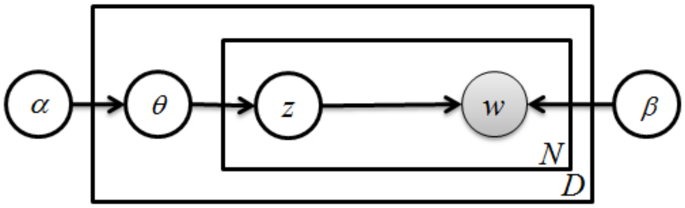


Figure 6.1: LDA

their ratings. *PLDA* learns one latent variable from opinion phrases. *S-PLDA* learns both aspects and their corresponding ratings from opinion phrases. Finally, *D-PLDA* learns aspects and ratings from opinion phrases while considering the dependency between the generated aspects and ratings.

We also present a novel technique for extracting opinion phrases from reviews based on dependency parsing. Different from current preprocessing methods which are mainly based on syntactic properties our technique is based on the semantic relationships between words. We conduct extensive experiments on a real life dataset from Epinions.com, and based on the results we propose a set of design guidelines for LDA models for aspect-based opinion mining.

6.2 LDA Models for Aspect-based Opinion Mining

In this section, we first briefly discuss the basic LDA model for the problem of aspect-based opinion mining. Then we introduce a series of models based on LDA making different probabilistic assumptions.

6.2.1 LDA

Latent Dirichlet Allocation (LDA) is a generative probabilistic model of a corpus [14]. The basic idea is that documents are represented as mixtures over latent topics where topics are associated with a distribution over the words of the vocabulary. Figure 6.1 shows the graphical model of this model. LDA assumes the following generative process:

1. Sample $\theta \sim Dir(\alpha)$.
2. For each word w_n, $n \in \{1, 2, ..., N\}$
   (a) Sample a topic $z_n \sim Mult(\theta)$

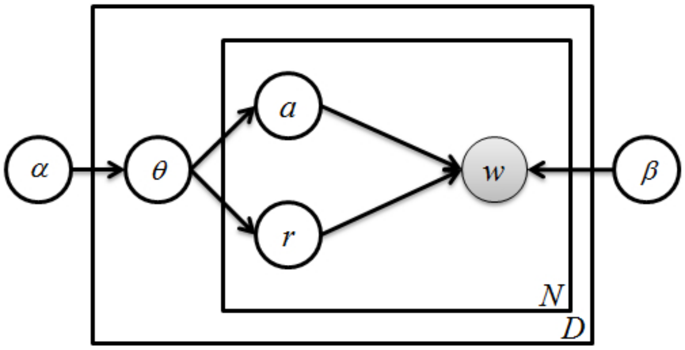


Figure 6.2: S-LDA

(b) Sample a word $w_n \sim P(w_n|z_n, \beta)$, a multinomial distribution conditioned on the topic z_n.

Translating this process into a joint probability distribution results in the following expression:

$$P(\boldsymbol{z}, \boldsymbol{w}, \theta|\alpha, \beta) = P(\theta|\alpha) \prod_{n=1}^{N} [P(z_n|\theta) P(w_n|z_n, \beta)] \tag{6.1}$$

Some of the current works [19, 194, 163, 164] apply this model on reviews to extract topics as aspects. In [19] and [194], the model has been used at the sentence level. The authors of [194] further improve the model by considering different word distributions for aspects, ratings, and background words. In [163, 164] the basic LDA model is improved by considering different topic distributions for local and global topics and is applied at the document level.

6.2.2 S-LDA

The second model replaces the one latent variable for topics by two separate variables for aspects and ratings. We call this model Separate-LDA (S-LDA). For every aspect/rating pair, θ contains the probability of generating that combination of aspect and rating. The variable θ is sampled once per review. After sampling θ, the latent variables a_n and r_n are sampled independently (conditional independency), and then a word w_n is sampled conditioned on the sampled aspect and rating (Figure 6.2). The joint probability distribution of this model is as follows:

$$P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{w}, \theta|\alpha, \beta) = P(\theta|\alpha) \prod_{n=1}^{N} [P(a_n|\theta) P(r_n|\theta) P(w_n|a_n, r_n, \beta)] \tag{6.2}$$

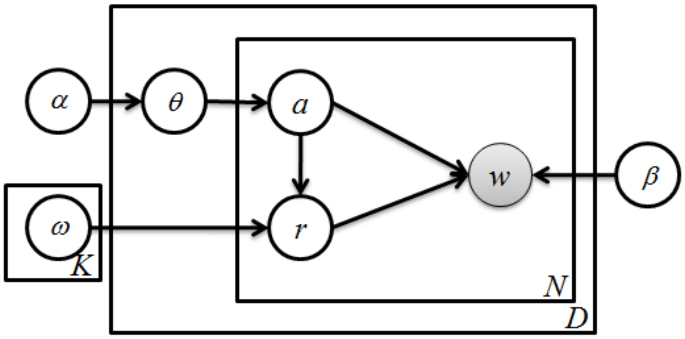


Figure 6.3: D-LDA

A model similar to S-LDA has been proposed in [77], learning two Dirichlet distributions (one for aspects and one for ratings) per review. While that model further considers the syntactic dependency between words and is more an LDA-HMM model, the generation of words conditioned on both aspects and ratings is the same as the generative process of S-LDA.

6.2.3 D-LDA

Dependency-LDA (D-LDA) also models the dependency between the latent aspects and ratings while learning from a bag-of-words model of reviews (Figure 6.3). The joint probability distribution of D-LDA considers this dependency:

$$P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{w}, \theta|\alpha, \beta, \boldsymbol{\omega}) = P(\theta|\alpha) \prod_{n=1}^{N} [P(a_n|\theta) P(r_n|a_n, \omega) P(w_n|a_n, r_n, \beta)] \tag{6.3}$$

Several models similar to D-LDA have been proposed in the literature [171, 46, 64]. The model proposed in [171] further considers the weight placed on each aspect by the reviewer. The model proposed in [46] assumes the dependency of the selected aspect from the sampled rating, i.e., the opposite direction of the dependency. The authors of [64] also make the same assumption as [46] and apply the model at the sentence level to extract aspects and ratings.

6.2.4 PLDA

While LDA assumes a bag-of-words model, Phrases-LDA (PLDA) assumes a bag-of-phrases model of reviews (Figure 6.4). A review is preprocessed into a bag-of-opinion-phrases $< h_n, m_n >$ which leads to two observed variables h_n (head term) and m_n (modifier). Translating this process into a

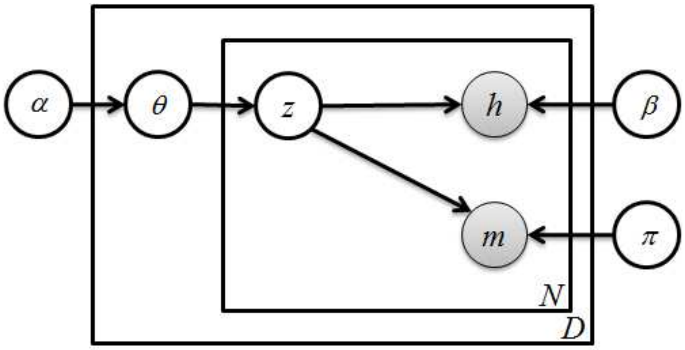


Figure 6.4: PLDA

joint probability distribution results in the expression:

$$P(\boldsymbol{z}, \boldsymbol{h}, \boldsymbol{m}, \theta|\alpha, \beta, \pi) = P(\theta|\alpha) \prod_{n=1}^{N} [P(z_n|\theta) P(h_n, m_n|z_n, \beta, \pi)] \tag{6.4}$$

In [188] a similar model is applied on opinion phrases to extract topics from reviews. While the generation of opinion phrases in [188] is the same as the generative process of PLDA, their model further assumes that each sentence of the review is related to only one topic.

6.2.5 S-PLDA

Compared to PLDA, the S-PLDA model introduces a separate rating variable which is conditionally independent from the aspect. In this model a review is assumed to be generated by first choosing a value of θ, and then repeatedly sampling N aspects and ratings as well as opinion phrases $< h_n, m_n >$ conditioned on the chosen value of θ. Similar to S-LDA, θ represents the aspect/rating pairs and for every pair, α contains the probability of generating that combination of aspect and rating (Figure 6.5). The joint probability distribution of S-PLDA is as follows:

$$P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \theta|\alpha, \beta, \pi) = P(\theta|\alpha) \prod_{n=1}^{N} [P(a_n|\theta) P(r_n|\theta) P(h_n|a_n, \beta) P(m_n|r_n, \pi)] \tag{6.5}$$

where $P(h_n|a_n, \beta)$ and $P(m_n|r_n, \pi)$ are multinomial distributions conditioned on the aspect a_n and rating r_n, respectively. The same model is presented in our previous work [111] as a comparison partner under the name of MLDA.

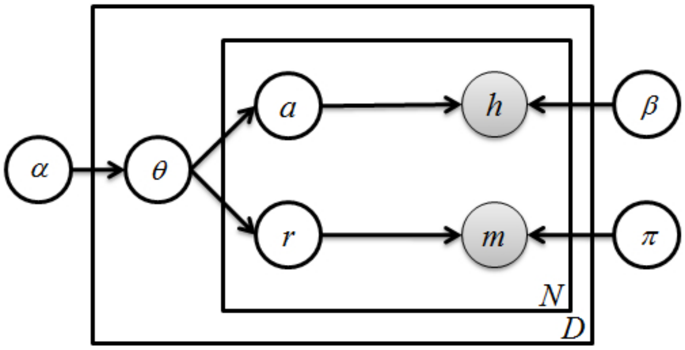


Figure 6.5: S-PLDA

6.2.6 D-PLDA

Similar to the step from S-LDA to D-LDA, compared to S-PLDA, the D-PLDA model adds the dependency between ratings and aspects. There are various options for dependencies between the two latent variables and the two observed variables. We assume that modifiers depend on the aspect and the rating. On the other hand, we assume that the rating of an aspect does not affect the choice of a head term for that aspect.

D-PLDA can be viewed as generative process that first generates an aspect and subsequently generates its rating. In particular, for generating an opinion phrase, this model first generates an aspect a_n from an LDA model. Then it generates a rating r_n conditioned on the sampled aspect a_n. Finally, a head term h_n is drawn conditioned on a_n and a modifier m_n is generated conditioned on both the aspect a_n and rating r_n (Figure 6.6). D-PLDA specifies the following joint distribution:

$$P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \theta | \alpha, \boldsymbol{\omega}, \beta, \boldsymbol{\pi}) = P(\theta|\alpha) \prod_{n=1}^{N} [P(a_n|\theta) P(r_n|a_n, \boldsymbol{\omega}) P(h_n|a_n, \beta) P(m_n|a_n, r_n, \boldsymbol{\pi})] \tag{6.6}$$

This model is presented in our previous work [111] under the name of ILDA.

6.3 Inference and Estimation

Computing the posterior distribution of the latent variables for the LDA models is intractable. Blei et al. [14] proposed to obtain a tractable lower bound by modifying the graphical model through

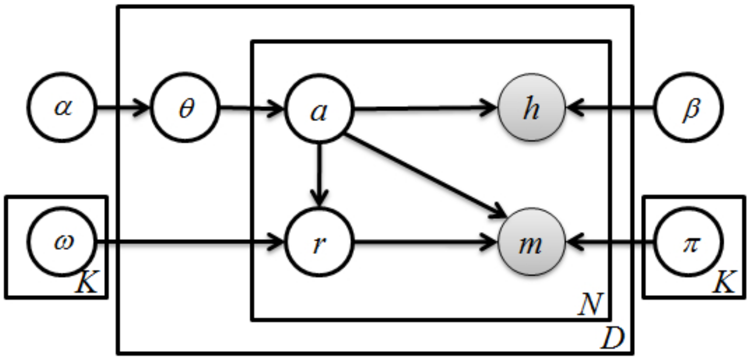


Figure 6.6: D-PLDA

considering a variational Dirichlet parameter for generating θ and a variational multinomial parameter for generating each latent variable. In a good approximation, the KL-divergence between the variational distribution and the true posterior will be minimal. So, by setting the derivative of the KL-divergence with respect to variational parameters equal to zero, the update equations can be obtained. Using Variational Estimation-Maximization (EM) technique [14], a lower bound on the posterior probability can be obtained.

Regarding the computational complexity, each iteration of variational inference for the basic LDA requires $O(Nk)$ operations [14] where k is the number of topics. According to the variational inference algorithms, S-LDA and D-LDA require $O(5Nk)$, PLDA and S-PLDA require $O(2Nk)$, and D-PLDA require $O(6Nk)$ operations for each iteration. As stated in [14], the number of iterations required for a single document is on the order of the number of words in the document. This means that the total number of operations for the LDA models is roughly $O(N^2k)$.

When working with conditional distributions, over-fitting is always a serious problem [14]. A new review is very likely to contain words that did not appear in any of the reviews in a training corpus. Maximum likelihood estimate of the model parameters assign zero probability to such words, and so zero probability to new reviews. Smoothing is a standard approach to dealing with this problem [14]. We smooth all the parameters which depend on the observed data by assigning positive probability to all vocabulary words whether or not they are observed in the training set.

Finally, when estimating model parameters using maximum likelihood estimation, it is possible to increase the likelihood by adding parameters, which may however result in over-fitting. Since our goal is to compare the average performance of different models, we perform our experiments for different values of $k = \{5, 10, 15, 20, 25\}$. Note that, before applying the models on reviews, we first apply the Porter Stemmer algorithm [133] and then remove stop words using a standard lists of stop words[2].

6.4 Extraction of Opinion Phrases

Bag-of-words is a popular representation of documents in text processing. In the area of aspect-based opinion mining most of the current works [19, 194, 163, 164, 77, 82, 88, 46, 171, 64] adopt this representation of reviews. However, it is not clear whether representing a review as a bag of words is sufficient for this problem. Some of the recent works [188, 111] propose to preprocess the reviews to extract opinion phrases and present LDA models that generate only opinion phrases. These works typically use some simple parsing techniques to extract pairs of frequent noun and nearest adjective, or use POS patterns, e.g. "_Adj _NN", "_NN _VB _Adj", etc.

In this section, we present a novel method for extracting opinion phrases based on the Stanford Dependency parser [21], which is a parser widely used in the area of text mining. A dependency parser determines the semantic relationships between words and promises to generate opinion phrases more accurately than methods that consider only the proximity of words. Dependency parsers provide a simple description of the grammatical relationships in a sentence. In the following, we briefly explain the grammatical relations [21] we use:

- Adjectival complement (*acomp*): An adjectival phrase which functions as the complement, e.g., "The auto-mode works amazing" parsed to 'acomp(works, amazing)'.

- Adjectival modifier (*amod*): An adjectival phrase that serves to modify the meaning of a noun phrase, e.g., "It has a wide screen" parsed to 'amod(screen, wide)';

- "And" conjunct (*conj_and*): A relation between two elements connected by the coordinating conjunction "and", e.g., "The LCD is small and blurry" parsed to 'conj_and(small, blurry)'.

- Copula (*cop*): A relation between the complement of a copular verb and the copular verb, e.g., "The batteries are ok" parsed to 'cop(ok, are)'.

- Direct object (*dobj*): A noun phrase which is the object of the verb, e.g., "I like the auto-focus" parsed to 'dobj(like, auto-focus)'.

[2] http://ir.dcs.gla.ac.uk/resources/linguistic_utils/stop_words

- Negation modifier (*neg*): A relation between a negation word and the word it modifies, e.g., “The shutter lag isn’t fast” parsed to ‘neg(fast, n’t)’.
- Noun compound modifier (*nn*): A noun that serves to modify the head noun, e.g., “The shutter lag is’n fast” parsed to ‘nn(lag, shutter)’.
- Nominal subject (*nsubj*): A noun phrase which is the syntactic subject of a clause, e.g., “The zoom is disappointing” parsed to ‘nsubj(disappointing, zoom)’.

We employ these grammatical relations to define a set of dependency patterns for extracting opinion phrases. In the following we list the extraction patterns (N indicates a noun, A an adjective, V a verb, h a head term, m a modifier, and $< h, m >$ an opinion phrase). Table 6.1 shows some examples of how these patterns are used for extracting opinion phrases.

1. $amod(N, A) \rightarrow < N, A >$
2. $acomp(V, A) + nsubj(V, N) \rightarrow < N, A >$
3. $cop(A, V) + nsubj(A, N) \rightarrow < N, A >$
4. $dobj(V, N) + nsubj(V, N') \rightarrow < N, V >$
5. $< h_1, m > + conj\_and(h_1, h_2) \rightarrow < h_2, m >$
6. $< h, m_1 > + conj\_and(m_1, m_2) \rightarrow < h, m_2 >$
7. $< h, m > + neg(m, not) \rightarrow < h, not + m >$
8. $< h, m > + nn(h, N) \rightarrow < N + h, m >$
9. $< h, m > + nn(N, h) \rightarrow < h + N, m >$

6.5 Experiments

In the next subsections, we first briefly describe the dataset we used and then present a qualitative and quantitative evaluation of the LDA models. For qualitative analysis we compare the top words obtained by different models. For the quantitative analysis we measure the performance of the models in terms of test set likelihood.

Table 6.1: Dependency patterns for extracting opinion phrases

Sentence	Dependency Relations	Patrn.	Opinion Phrases
This camera has great zoom and resolution.	amod(zoom,great), conj_and(zoom, resolution)	1, 5	<zoom,great>, <resolution, great>
It comes with small and rechargeable batteries.	amod(batteries,rechargeable), conj_and(small, rechargeable)	1, 6	<batteries,rechargeable>, <batteries, small>
The camera case looks nice.	acomp(looks, nice), nsubj(looks, case), nn(case, camera)	2, 8	<camera case, nice>
I love the picture quality.	dobj(love, picture), nsubj(love, I), nn(quality, picture)	4, 9	<picture quality, love>
The screen is wide and clear.	cop(wide, is), nsubj(wide, screen), conj_and(wide, clear)	3, 6	<screen,wide>, <screen, clear>
The battery life is not long.	cop(long, is), nsubj(long, life), nn(life, battery), neg(long, not)	3,8,7	<battery life, not long>

6.5.1 Dataset

We built a crawler to extract product reviews from the well-known reviewing website Epinions.com. The dataset contains 505,978 reviews about 94,792 products from 257 product categories (e.g., camcorder, cellular phone, digital camera, Mp3 player, etc.). Note that, in aspect-based opinion mining, since both aspects and ratings are item-specific, one model is learned per item. Out of 94,792 products, 49,324 products have only one review which makes it impossible to train and test, so these products were removed.

To the best of our knowledge, the impact of the size of the training dataset has not been evaluated in the literature on aspect-based opinion mining. To do so, we define five subsets of products with different numbers of reviews. The first subset contains products with at least 2 and at most 10 reviews. We also consider subsets of products with more than 10 and less than 50 reviews, between 50 and 100 reviews, from 100 to 200 reviews, and more than 200 reviews. For each subset of products, Table 6.2 shows the number of products in that subset and the average number of reviews per product. In the following sections we present evaluation results for each subset of products as the average of the results for the products of that subset.

Table 6.2: Statistics of the dataset

Subset	#Products	#Rev./Product
$1 < \#Rev. <= 10$	36,166	3
$10 < \#Rev. <= 50$	7,886	19
$50 < \#Rev. <= 100$	869	67
$100 < \#Rev. <= 200$	368	137
$200 < \#Rev.$	179	341

6.5.2 Qualitative Evaluation

To perform a qualitative evaluation, we select a product from the digital camera category that has 166 reviews. Table 6.3 shows the top (most probable) words/phrases extracted by different models for this product. We also compare the performance of models learning from phrases (PLDA, S-PLDA, and D-PLDA) using different preprocessing techniques:

- Frequent noun technique: Pairs of frequent nouns and nearest adjectives [111].
- POS patterns: Pairs of nouns and adjectives extracted using POS patterns [188].
- Dependency patterns (introduced in Section 6.4)

By comparing the top words of the first three models, we observe that the extracted words are almost the same, i.e., S-LDA and D-LDA do not perform better than the basic LDA. Comparing the extracted phrases using frequent noun technique and POS patterns, we see that the frequent noun technique is not that promising since there are some frequent phrases which are not relevant (e.g., <time, hard> and <pictur, mani>) and there are lots of opinion phrases which are not frequent (e.g., <pictur, good> and <displai, nice>). It is also shown that often the extracted phrases based on dependency patterns are more informative, e.g., <qualiti, amaz>, <view find, love>, and <photo qualiti, high>.

6.5.3 Quantitative Evaluation

If the necessary ground truth is available, the performance of a model can be evaluated by measures such as accuracy, precision and recall. However, in large data sets such as our Epinions dataset ground truth is typically not available. In such cases, a standard approach for the evaluation of

Table 6.3: Top words/phrases extracted by different LDA models

Prep.	Model	Top words/phrases extracted for a digital camera (stemmed)
N/A	LDA	good, pictur, digit, resolut, set, disk, great, time, shot, featur
	S-LDA	featur, zoom, disk, good, pictur, set, shot, resolut, camera, floppi
	D-LDA	bright, time, resolut, good, disk, set, great, digit, shot, camera
Freq. nouns	PLDA	<pictur, mani>, <resolut, high>, <time, hard>, <camera, digit>, <disk, floppi>
	S-PLDA	<time, hard>, <drive, floppi>, <resolut, mani>, <camera, digit>, <featur, good>
	D-PLDA	<resolut, high>, <camera, digit>, <drive, floppi>, <usb, easi>, <disk, hard>
POS patrn.	PLDA	<usag, normal>, <price, high>, <drive, floppi>, <pictur, good>, <featur, sever>
	S-PLDA	<pictur, good>, <effect, special>, <displai, nice>, <life, long>, <printer, great>
	D-PLDA	<batteri, dead>, <resolut, mani>, <camera, digit>, <pictur, good>, <featur, offer>
Dep. patrn.	PLDA	<displai, nice>, <zoom, optic>, <effect, mani>, <pictur, good>, <reolut, high>
	S-PLDA	<resolut, high>, <qualiti, amaz>, <printer, compat>, <displai, nice>, <zoom, optic>
	D-PLDA	<photo qualiti, high>, <zoom, optic>, <storag capac, unlimit>, <printer, compat>, <viewfind, love>

graphical models is comparing the likelihoods of a held-out test set. We hold out 10% of the reviews for testing purposes and use the remaining 90% to train the model. As is standard for LDA models [14, 188, 111], we computed the perplexity of the held-out test set for all models for various numbers of aspects, $k = \{5, 10, 15, 20, 25\}$. Since the relative results are similar for different values of k, we choose $k = 15$ for our discussion. The perplexity is monotonically decreasing in the likelihood of the test data, and a lower perplexity score indicates better performance.

While perplexity is a well-established measure for comparing LDA models, it is not clear how perplexity relates to the accuracy of aspect identification and rating prediction. To this end, we use a well-known public dataset with ground truth used in [48, 49, 30] (with 314 reviews of 5 products) to analyze the correlation between model perplexity and accuracy of these tasks. Table 6.4 shows the average precision and recall of aspect identification, the Mean Squared Error (MSE) of rating prediction and the perplexity of different models on this dataset averaged over all products.

Table 6.4: Evaluation on labeled dataset

Model	Precision	Recall	MSE	Perplexity
S-LDA	0.54	0.51	1.25	813.11
LDA	0.54	0.52	1.22	795.72
D-LDA	0.58	0.55	1.18	748.26
PLDA	0.81	0.73	0.96	587.82
S-PLDA	0.83	0.73	0.93	335.02
D-PLDA	0.87	0.78	0.85	131.80

We observe a strong correlation of the perplexity and precision, recall, and MSE. All three accuracy measures improve monotonically with improving (decreasing) perplexity. Extrapolating these results to the much larger Epinions dataset, for which we have no ground truth, we argue that the perplexity results reported in our quantitative evaluation provide a good indication of the relative accuracy of aspect identification and rating prediction for the compared models.

Comparing Preprocessing Techniques

We compare the perplexity of models learning from opinion phrases for different preprocessing techniques. As the baselines we use pairs of one frequent noun and one adjective [111] and phrases generated from POS patterns [188].

Which preprocessing technique for extracting opinion phrases works best? To answer this question, Table 6.5 presents the average perplexity of different models using different preprocessing techniques. The results indicate that using POS patterns for extracting opinion phrases is more effective than the frequent noun technique. However, it is unclear whether this better performance is due the infrequency of some of the opinion phrases or because of the inaccuracy of extracted frequent phrases. Table 6.5 also demonstrates that our proposed technique based on dependency parsing clearly and consistently outperforms the other preprocessing techniques for all subsets of products. This confirms our hypothesis that exploiting the semantic relationship between words pays off for extracting opinion phrases.

Table 6.5: Perplexity of the LDA models using different preprocessing techniques

(a) Frequency nouns

Subset of Products	PLDA	S-PLDA	D-PLDA
$1 < \#Rev. <= 10$	11834.99	11824.35	11740.95
$10 < \#Rev. <= 50$	6724.61	6129.57	5829.33
$50 < \#Rev. <= 100$	3024.59	2243.22	1882.91
$100 < \#Rev. <= 200$	1885.52	1511.47	660.96
$200 < \#Rev.$	1337.93	1165.88	406.27

(b) POS patterns

Subset of Products	PLDA	S-PLDA	D-PLDA
$1 < \#Rev. <= 10$	7735.32	7679.10	7650.67
$10 < \#Rev. <= 50$	4573.20	4124.06	3847.92
$50 < \#Rev. <= 100$	2734.58	1912.88	1381.86
$100 < \#Rev. <= 200$	1753.39	1328.77	411.94
$200 < \#Rev.$	1301.00	1066.89	353.73

(c) Dependency patterns

Subset of Products	PLDA	S-PLDA	D-PLDA
$1 < \#Rev. <= 10$	5463.80	5422.24	5413.66
$10 < \#Rev. <= 50$	3438.19	2937.36	1975.35
$50 < \#Rev. <= 100$	1998.42	1514.71	592.40
$100 < \#Rev. <= 200$	1481.00	1038.25	164.19
$200 < \#Rev.$	1284.19	879.29	142.37

Evaluation of LDA Models

In this section we compare the performance of the discussed models to answer the key design questions stated in the introduction. Figure 6.7(a) shows the perplexity of all models for different subsets of products. The perplexity of the models learning from opinion phrases are given for preprocessing using dependency patterns, which performs best according to Table 6.5. Figure 6.7 compares only products with at least 10 reviews, similar to the literature. At the end of this section, we also discuss the models' performance for products with less than 10 reviews.

Is it better to have separate latent variables for aspects and ratings? To answer this question, we compare the performance of LDA vs. S-LDA (Figure 6.7(b)), and also PLDA vs. S-PLDA (Figure 6.7(c)). It turns out that having separate latent variables for aspects and ratings cannot improve the performance of a model having only one observed variable (i.e., using the bag-of-words model). As a result, the performance of LDA and S-LDA are almost the same. However, S-PLDA outperforms PLDA thanks to the separate aspect and rating variables generating head terms and modifiers, respectively.

Is it better to assume dependency between ratings and aspects? Here we compare the perplexity of S-LDA vs. D-LDA (Figure 6.7(d)) and also S-PLDA vs. D-PLDA (Figure 6.7(e)). The higher perplexity of D-LDA compared to S-LDA demonstrates that increasing model complexity while keeping the observed data fixed decreases the performance of the model. However, the comparison of D-PLDA and S-PLDA indicates that assuming the dependency between ratings and aspects improves the performance of a model that generates opinion phrases.

Is it better to learn from bag-of-words or preprocess the reviews and learn from opinion phrases? To address this question, we compare three pairs of models: LDA vs. PLDA (Figure 6.7(f)), S-LDA vs. S-PLDA (Figures 6.7(g)), and D-LDA vs. D-PLDA (Figure 6.7(h)). The only difference between these pairs of models is generation of words vs. generation of opinion phrases. Except for products with more than 200 reviews, LDA performs better than PLDA, showing that extracting opinion phrases does not help if only one latent variable is used to generate both head terms and modifiers. Comparing S-LDA and S-PLDA, we observe that for products with more than 50 reviews S-PLDA achieves a lower perplexity (better performance), while for products with fewer reviews it performs poorer. Finally, D-PLDA performs significantly better than D-LDA in all of the product subsets, showing the advantage of learning from opinion phrases rather than bag-of-words when modeling aspects, ratings, and their dependency.

Does the answer to the above questions differ for products with few reviews and products with many reviews? Here, we discuss the performance of the LDA models for products with less than 10 reviews. Since products of this subset have only 3 reviews on average (see Table 6.2), learning a proper model is very difficult. As Table 6.6 shows, the basic LDA model outperforms the more complex models for these products, demonstrating that neither preprocessing nor model complexity can improve the performance of LDA. Thus, the best design choices are very different for products with many reviews and for those with few reviews.

6.6 Conclusion

LDA-based models are considered to be state-of-the-art for aspect-based opinion mining. Realizing that there is no "one-size-fits-all" model that always outperforms all other models, in this work we

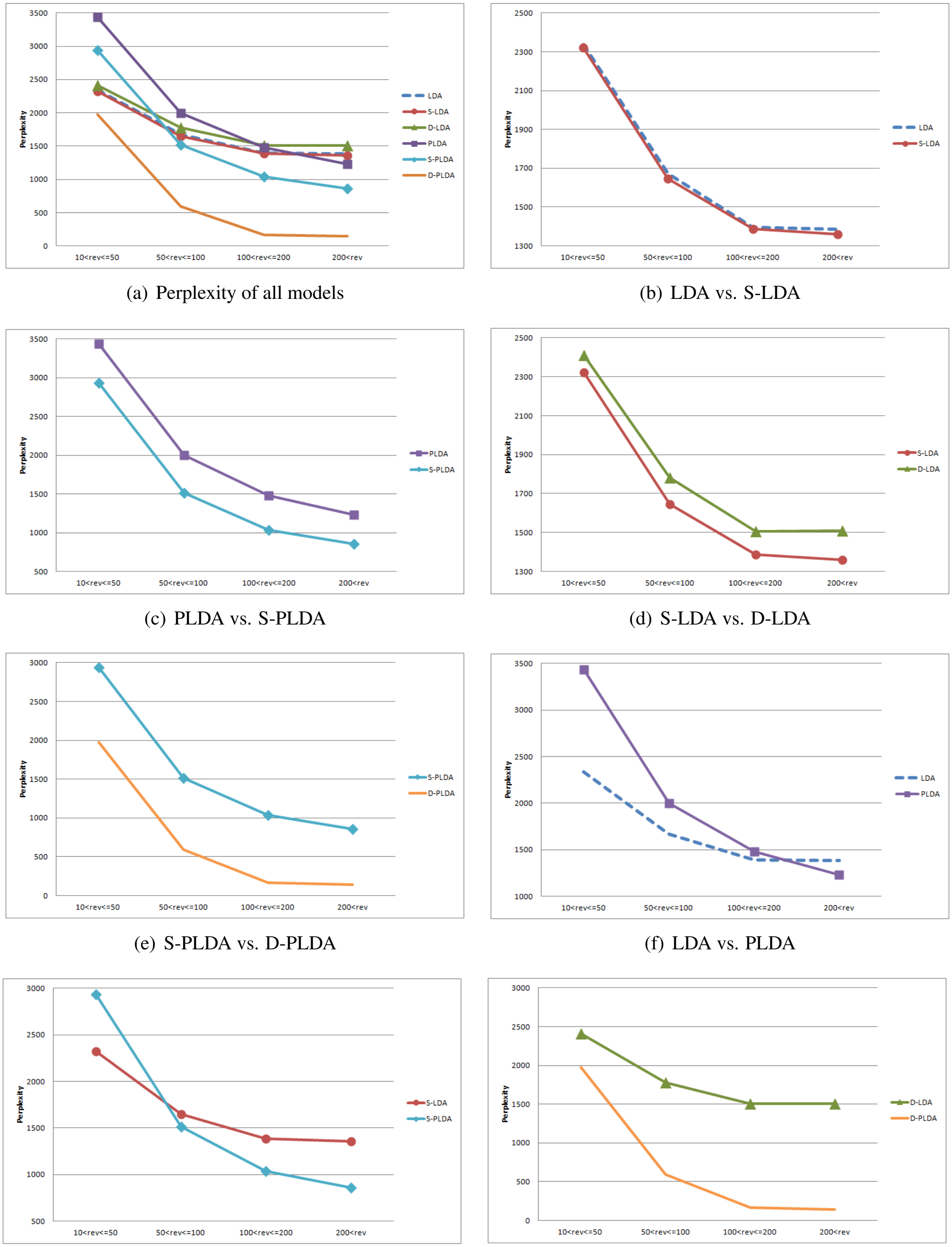
Table 6.6: Perplexity of products with $\#Rev. <= 10$

LDA	S-LDA	D-LDA	PLDA	S-PLDA	D-PLDA
4413.6	4567.2	4729.3	5463.8	5422.2	5413.6

developed a set of design guidelines. We conducted extensive experiments on a very large real life dataset from Epinions.com (500K reviews) and compared the performance of different models in terms of the likelihood of the held-out test set. Based on our experimental results, we formulate the following guidelines for the design of LDA models for aspect-based opinion mining:

- When learning from bag-of-words, having separate latent variables for aspects and ratings cannot improve the performance of a model. However, when learning from opinion phrases, it does help to consider two latent variables for generating head terms and modifiers.
- When learning from opinion phrases and having separate latent variables for aspect and rating, assuming their dependency improves the performance.
- When separate latent variables are assumed for aspects and ratings, using preprocessing techniques can improve the performance.
- Using dependency patterns consistently achieves the best performance for extracting opinion phrases.
- For products with few reviews, the basic LDA model outperforms the more complex models. For products with many reviews, the model learning aspects and ratings from opinion phrases with dependency assumption performs best.

This work suggests several directions for future research, such as investigating different factors for improving the model performance for items with only few reviews. Another direction is exploring the impact of different additional input sources (e.g. review's overall rating) on the performance of the models. In the next chapter, we propose a model based on D-PLDA (ILDA) to improve the performance for items with few reviews.



(a) Perplexity of all models

(b) LDA vs. S-LDA

(c) PLDA vs. S-PLDA

(d) S-LDA vs. D-LDA

(e) S-PLDA vs. D-PLDA

(f) LDA vs. PLDA

(g) S-LDA vs. S-PLDA

(h) D-LDA vs. D-PLDA

Figure 6.7: Perplexity comparisons for different subsets of products

Chapter 7

FLDA: Addressing the Cold Start Problem

In the last decade, several latent variable models have been proposed to address the problem of aspect-based opinion mining, e.g., [163, 194, 19, 188, 58, 101, 42, 88, 111]. All of these models are applied at the item level, i.e., they learn one model per item from the reviews of that item. Learning a model per item is logical as the rating of an aspect depends on the aspect quality which usually differs for different items. However, an issue that has been neglected in all of the current works is that latent variable models are not accurate if there is not enough training data. In our previous work (Chapter 6), we evaluated the impact of the size of the training dataset on a series of LDA-based models for aspect-based opinion mining. Our comprehensive evaluation of these models on a real-life data set proved that while item level models work well for items with large number of reviews, they perform poorly when the size of the training dataset is small. In fact, the experimental evaluation showed that the basic LDA model outperforms the more complex models for these items. Borrowing a term from the recommender systems literature, we call such items *cold start items*. In real-life data sets such as those from Epinions.com and Amazon.com more than 90% of items are cold start (less than 10 reviews) which indicates there is a great need for accurate opinion mining models for these items.

In this work, we introduce the problem of identifying aspects and estimating their ratings for cold start items. To address this problem, we propose a probabilistic graphical model based on LDA, called *Factorized LDA (FLDA)*. The underlying assumption of this model is that the aspects and corresponding ratings of reviews are influenced not only by the items but also by the reviewers.

It further assumes that both items and reviewers can be modeled by a set of latent factors. Item factors represent the item's probability distribution over aspects and for each aspect its distribution over ratings. In the same way, reviewer factors represent the reviewer's probability distribution over aspects and for each aspect its distribution over ratings. FLDA generates aspects and ratings of reviews by learning the latent factors of items and reviewers.

Different from state-of-the-art LDA models which are learned per item, FLDA is trained at the category level. Note that, a category of items is a set of items sharing common characteristics, e.g., MP3 players, scanners, Bed and Breakfast Inns, etc. FLDA generates each aspect of a review based on both the aspect distribution of the corresponding item and the aspect distribution of the reviewer. It further generates the rating of an aspect depending on that aspect, the rating distribution of that aspect for that item and the rating distribution of that aspect for the reviewer. These distributions are trained using the reviews of all the items of a category, in particular the non cold start items, and serve as prior for the distributions of cold start items that otherwise could not be learned accurately. In other words, for cold start items the aspect distribution is mainly determined by the prior aspect distribution of the category, and the rating distribution of an aspect is mainly determined by the rating distribution of the reviewer or by the prior rating distribution of all reviewers (if the reviewer is cold start, i.e., has written few reviews). On the other hand, for non-cold start items the aspect and rating distributions are mainly determined by the observed reviews of that item.

We report the results of our extensive experiments on three real-life datasets from Epinions, Amazon, and TripAdvisor. The results demonstrate the improved effectiveness of the FLDA model in terms of likelihood of the held-out test set, in particular for cold start items. We also evaluate the accuracy of FLDA based on two application-oriented measures: item categorization and overall rating prediction for reviews. Both applications are performed based on the learned latent factors. We evaluate these applications by comparing the accuracy of the learned classifiers with the state-of-the-art techniques.

The remainder of the chapter is organized as follows. In the next section we will briefly discuss our contribution. Section 7.2 presents the proposed model, FLDA. Section 7.3 describes the inference and estimation techniques for FLDA. In Sections 7.4 and 7.5, we report the results of our experimental evaluation and discuss two applications of our model. Finally, Section 7.6 concludes the chapter with a summary and the discussion of future work.

7.1 Our Contribution

As discussed in the introduction, all of the current aspect-based opinion mining models are at the item level. However, learning a model at the item level is not accurate for cold start items, i.e., items that have been reviewed by few reviewers. Since a very large portion of items in real-life reviewing websites are cold start, having a proper model for these items is essential. To address the problem of aspect-based opinion mining for cold start items, we propose a probabilistic model based on LDA, called FLDA. This model assumes that both items and reviewers can be modeled by a set of latent factors. Item's/reviewer's factors represent the item/reviewer distribution over aspects and for each aspect its distribution over ratings. Each review in the FLDA model is generated based on the learned factors of the corresponding item and reviewer. It first samples aspects in a review from the aspect distributions of the corresponding item and reviewer, and then generates the rating of each aspect conditioned on that aspect and the rating distributions of that item and reviewer. For cold start items, the aspect and rating distributions are mainly determined by the prior aspect distribution of the category and the rating distribution of the reviewer (or the prior rating distribution of all reviewers), respectively. For non cold start items, the aspect and rating distributions mainly depend on the observed reviews of that item. In the following section, we will elaborate the proposed FLDA model in detail.

7.2 FLDA: Factorized Latent Dirichlet Allocation

In this work, we introduce a probabilistic model based on LDA, called Factorized LDA (FLDA), which models not only items but also reviewers. The FLDA model makes the following assumptions:

- A category has a set of aspects which are shared by all items in that category. For example, {zoom, battery life, shutter lag, etc.} is a set of aspects shared by all products in the category 'digital camera'. Note that, probabilities of occurrence of aspects can differ for different items in the category.
- Each item has a distribution over the aspects representing what aspects of its category are mainly commented on in reviews of that item. Each of these aspects is associated with a distribution of ratings.

- Each reviewer has a distribution over the aspects representing what aspects are more commented on by the reviewer. The reviewer is also associated, for each aspect, with a rating distribution.

Based on the above assumptions, to generate a review, aspects are first sampled conditioned on the aspect distributions of the corresponding item and reviewer. The rating of each aspect is then sampled conditioned on the aspect and the rating distributions of the item and the reviewer. Finally, opinion phrases are sampled based on the chosen aspects and ratings. Figure 7.1 shows the corresponding graphical model. As it is shown, the proposed model is built based on LDA which allows learning the aspect and rating priors. The other option for modeling the reviews is extending the PLSI model. However, PLSI does not learn the prior parameters and therefore cannot be used in our problem.

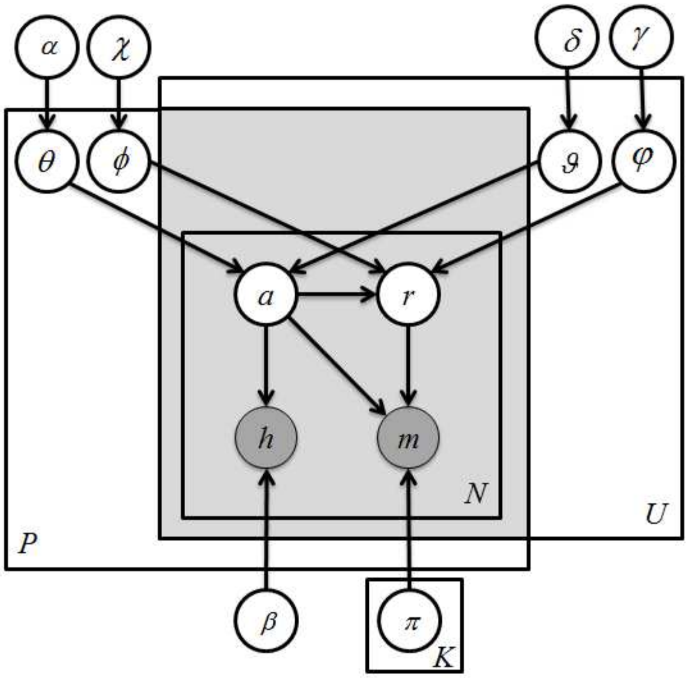


Figure 7.1: The graphical model for FLDA

As shown in Figure 7.1, α and δ are the prior aspect distributions and χ and γ are the prior rating distributions for the given category. The basic idea of FLDA is that each item p is represented as random mixtures over latent aspects, θ_p, and latent rating, ϕ_p, and each reviewer u is represented as random mixtures over latent aspect, ϑ_u, and latent ratings, φ_u. The FLDA model assumes the following generative process:

1. For each item p, $p \in \{1, 2, ..., P\}$
   (a) Sample $\theta_p \sim Dir(\alpha)$
   (b) Sample $\phi_p \sim Dir(\chi)$
2. For each reviewer u, $u \in \{1, 2, ..., U\}$
   (a) Sample $\vartheta_u \sim Dir(\delta)$
   (b) Sample $\varphi_u \sim Dir(\gamma)$
3. If there is a review by u about p, then for each opinion phrase $< h_{pun}, m_{pun} >$, $n \in \{1, 2, ..., N\}$
   (a) Sample $a_{pun} \sim P(a_{pun}|\theta_p, \vartheta_u)$ and sample $r_{pun} \sim P(r_{pun}|a_{pun}, \phi_p, \varphi_u)$
   (b) Sample $h_{pun} \sim P(h_{pun}|a_{pun}, \beta)$ and sample $m_{pun} \sim P(m_{pun}|a_{pun}, r_{pun}, \boldsymbol{\pi})$

where $P(h_{pun}|a_{pun}, \beta)$ and $P(m_{pun}|a_{pun}, r_{pun}, \boldsymbol{\pi})$ are multinomial distributions. In the following the resulting joint distribution of the FLDA model is presented:

$$P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\vartheta}, \boldsymbol{\varphi}|\alpha, \chi, \delta, \gamma, \beta, \boldsymbol{\pi}) = \prod_{p=1}^{P}[P(\theta_p|\alpha)P(\phi_p|\chi)] \prod_{u=1}^{U}[P(\vartheta_u|\delta)P(\varphi_u|\gamma)] \prod_{p=1}^{P}\prod_{u=1}^{U}\epsilon(p, u) \prod_{n=1}^{N}[P(a_{pun}|\theta_p, \vartheta_u) P(r_{pun}|a_{pun}, \phi_p, \varphi_u)P(h_{pun}|a_{pun}, \beta)P(m_{pun}|a_{pun}, r_{pun}, \boldsymbol{\pi})] \quad (7.1)$$

where $\epsilon(p, u) = 1$ if there is a review written by u about item p, otherwise $\epsilon(p, u) = 0$. The goal is to compute the posterior distribution of the latent variables given a review:

$$P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\vartheta}, \boldsymbol{\varphi}|\boldsymbol{h}, \boldsymbol{m}, \alpha, \chi, \delta, \gamma, \beta, \boldsymbol{\pi}) = \frac{P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\vartheta}, \boldsymbol{\varphi}|\alpha, \chi, \delta, \gamma, \beta, \boldsymbol{\pi})}{P(\boldsymbol{h}, \boldsymbol{m}|\alpha, \chi, \delta, \gamma, \beta, \boldsymbol{\pi})} \quad (7.2)$$

Similar to the basic LDA, due to the coupling between $\boldsymbol{\theta}$ and $\boldsymbol{\vartheta}$ with β and also between $\boldsymbol{\phi}$ and $\boldsymbol{\varphi}$ with $\boldsymbol{\pi}$, the conditional distribution of latent variables given observed data is intractable to compute. A wide variety of approximate inference algorithms have been proposed for LDA models. In this chapter, we use variational inference [14] to compute an approximation for the posterior distribution.

7.3 Inference and Parameter Learning

In this section, we describe approximate inference and parameter learning for the FLDA model, adopting a variational method. As computing the posterior distribution of the latent variables for FLDA is intractable, we obtain a tractable lower bound by modifying the graphical model through considering a variational parameters for generating each latent variable. In particular, we simplify FLDA into the graphical model shown in Figure 7.2.

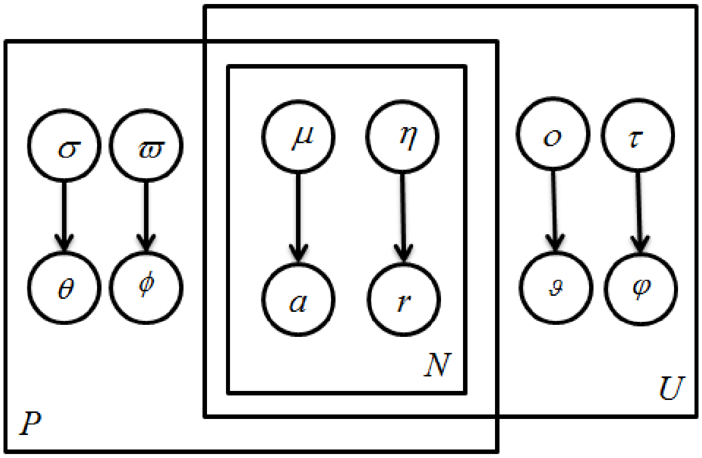


Figure 7.2: Graphical model representation of variational distribution for FLDA

This model specifies the following variational distribution on the latent variables:

$$Q(\boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\vartheta}, \boldsymbol{\varphi}, \boldsymbol{a}, \boldsymbol{r} | \boldsymbol{\sigma}, \boldsymbol{\varpi}, \boldsymbol{o}, \boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\eta}) = \prod_{p=1}^{P} [Q(\theta_p|\sigma_p) Q(\phi_p|\varpi_p)] \prod_{u=1}^{U} [Q(\vartheta_u|o_u) Q(\varphi_u|\tau_u)] \prod_{p=1}^{P} \prod_{u=1}^{U} \epsilon(p,u) \prod_{n=1}^{N} [Q(a_{pun}|\mu_{pun}) Q(r_{pun}|\eta_{pun})] \quad (7.3)$$

where the Dirichlet parameters $\boldsymbol{\sigma}$, $\boldsymbol{\varpi}$, $\boldsymbol{o}$ and $\boldsymbol{\tau}$, and the multinomial parameters $\boldsymbol{\mu}$ and $\boldsymbol{\eta}$ are free variational parameters. The KL-divergence between the variational distribution and the true posterior should be minimum to have a good approximation. To this end, we set the derivative of the KL-divergence with respect to variational parameters equal to zero, to obtain the update equations. The update equations are invoked repeatedly until the change in KL-divergence is small.

Algorithm 2 presents the pseudo-code of the variational inference procedure. In this algorithm β_{ix} is $P(h^x_{pun} = 1|a^i_{pun} = 1)$ for the appropriate x and π_{ijy} is $P(m^y_{pun} = 1|a^i_{pun} = 1, r^j_{pun} = 1)$ for the appropriate y. Recall that h_{pun} and m_{pun} are vectors with exactly one component equal to one. We can select the unique x and y such that $h^x_{pun} = 1$ and $m^y_{pun} = 1$ [14].

Algorithm 2 E-step of Variational Inference for FLDA

```
initialize $\mu^0_{puni} = 1/k$ for all $p$, $u$, $n$ and $i$
initialize $\eta^0_{punj} = 1/5$ for all $p$, $u$, $n$ and $j$
initialize $\sigma^0_{pi} = \alpha_i + (N \times U)/k$ for all $p$ and $i$
initialize $\varpi^0_{pij} = \chi_{ij} + (N \times U)/(k \times 5)$ for all $p$, $i$, $j$
initialize $o^0_{ui} = \delta_i + (N \times P)/k$ for all $u$ and $i$
initialize $\tau^0_{uij} = \gamma_{ij} + (N \times P)/(k \times 5)$ for all $u$, $i$, $j$
repeat
  for $p = 1$ to $P$ do
    for $u = 1$ to $U$ do
      if $\epsilon(p, u) == 1$ then
        for $n = 1$ to $N$ do
          for $i = 1$ to $k$ do
            $\mu^{t+1}_{puni} = \beta_{ix} \prod_j^5 \pi_{ijy}^{\eta^t_{punj}} \exp(\psi(\sigma^t_{pi})\psi(o^t_{ui}) + \sum_j^5 \eta^t_{punj}\psi(\tau^t_{uij})\psi(\varpi^t_{pij}))$
          end for
          normalize $\mu^{t+1}_{puni}$ to sum to 1
          for $j = 1$ to $5$ do
            $\eta^{t+1}_{punj} = \prod_i^K \pi_{ijy}^{\mu^t_{puni}} \exp(\sum_i^K \mu^t_{puni}\psi(\tau^t_{uij})\psi(\varpi^t_{pij}))$
          end for
          normalize $\eta^{t+1}_{punj}$ to sum to 1
        end for
      end if
    end for
  end for
  for $p = 1$ to $P$ do
    $\sigma^{t+1}_p = \alpha + \sum_u^U \sum_n^N \mu^{t+1}_{pun}\psi(o^{t+1}_u)$
    $\varpi^{t+1}_p = \chi + \sum_u^U \sum_n^N \mu^{t+1}_{pun}\eta^{t+1}_{pun}\psi(\tau^{t+1}_u)$
  end for
  for $u = 1$ to $U$ do
    $o^{t+1}_u = \delta + \sum_p^P \sum_n^N \mu^{t+1}_{pun}\psi(\sigma^{t+1}_p)$
    $\tau^{t+1}_u = \gamma + \sum_p^P \sum_n^N \mu^{t+1}_{pun}\eta^{t+1}_{pun}\psi(\varpi^{t+1}_p)$
  end for
until convergence
```

By computing the approximate posterior, we can find a lower bound on the joint probability,

Algorithm 3 M-Step of Variational Inference for FLDA

```
$\beta_{ix} = \sum_p^P \sum_u^U \sum_n^N \mu^*_{puni} h^x_{pun}$
$\pi_{ijy} = \sum_p^P \sum_u^U \sum_n^N \mu^*_{puni} \eta^*_{punj} m^y_{pun}$
$\alpha_{new} = \alpha_{old} - H(\alpha_{old})^{-1} g(\alpha_{old})$
$\chi_{new} = \chi_{old} - H(\chi_{old})^{-1} g(\chi_{old})$
$\delta_{new} = \delta_{old} - H(\delta_{old})^{-1} g(\delta_{old})$
$\gamma_{new} = \gamma_{old} - H(\gamma_{old})^{-1} g(\gamma_{old})$
```

$P(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\vartheta}, \boldsymbol{\varphi})$. Using this lower bound we can find approximate estimates for FLDA parameters via an alternative variational EM procedure [14]. The variational EM algorithm alternates between Expectation (E-step) and Maximization (M-step) steps until the bound on the expected log likelihood converges. The variational EM algorithm for FLDA is as follows[1]:

1. (E-step) For each review, find the optimizing values of the variational parameters σ^*, ϖ^*, o^*, τ^*, μ^*, and η^* (using Algorithm 2).

2. (M-step) Maximize the resulting lower bound on the log likelihood with respect to the model parameters α, χ, δ, γ, β, and $\boldsymbol{\pi}$ (using Algorithm 3).

The M-step update for the Dirichlet parameters α, χ, δ and γ are implemented using the Newton-Raphson optimization technique that finds a stationary point of a function by iterating [14]. In Algorithm 3, $H(x)$ and $g(x)$ are the Hessian matrix and gradient respectively at the point x.

Note that, to deal with over fitting, we smooth all the parameters which depend on the observed data by assigning positive probability to all vocabulary words whether or not they are observed in the training set.

7.4 Experiments

In this section, we first briefly describe the real-life datasets we used for our experiments and then present the results of the experimental evaluation of the FLDA model. We evaluate the performance of the model in terms of likelihood of the held-out test set and also based on two application-oriented measures for categorizing items and predicting reviews overall ratings.

[1] The detailed derivation of the variational EM algorithm is available at `http://http://www.sfu.ca/~sam39/FLDA/`

7.4.1 Datasets

To evaluate the proposed model, we performed experiments on three real-life datasets from Epinions [113], Amazon [62], and TripAdvisor [170]. In each dataset, we select items with at least one review. For preprocessing, we adopt the dependency pattern technique to identify opinion phrases in the form of a pair of head term and modifier. This technique results in the best performance in compare to other preprocessing techniques according to our previous work (Chapter 6). In Tables 7.1, general statistics of these datasets are shown.

Table 7.1: General statistics of different datasets

Dataset	Epinions	Amazon	TripAdvisor
#Categories	379	38	5
#Reviews	541,219	5,016,492	181,395
#Reviewers	109,857	1,761,879	117,976
#Items	87,633	1,108,018	1,496

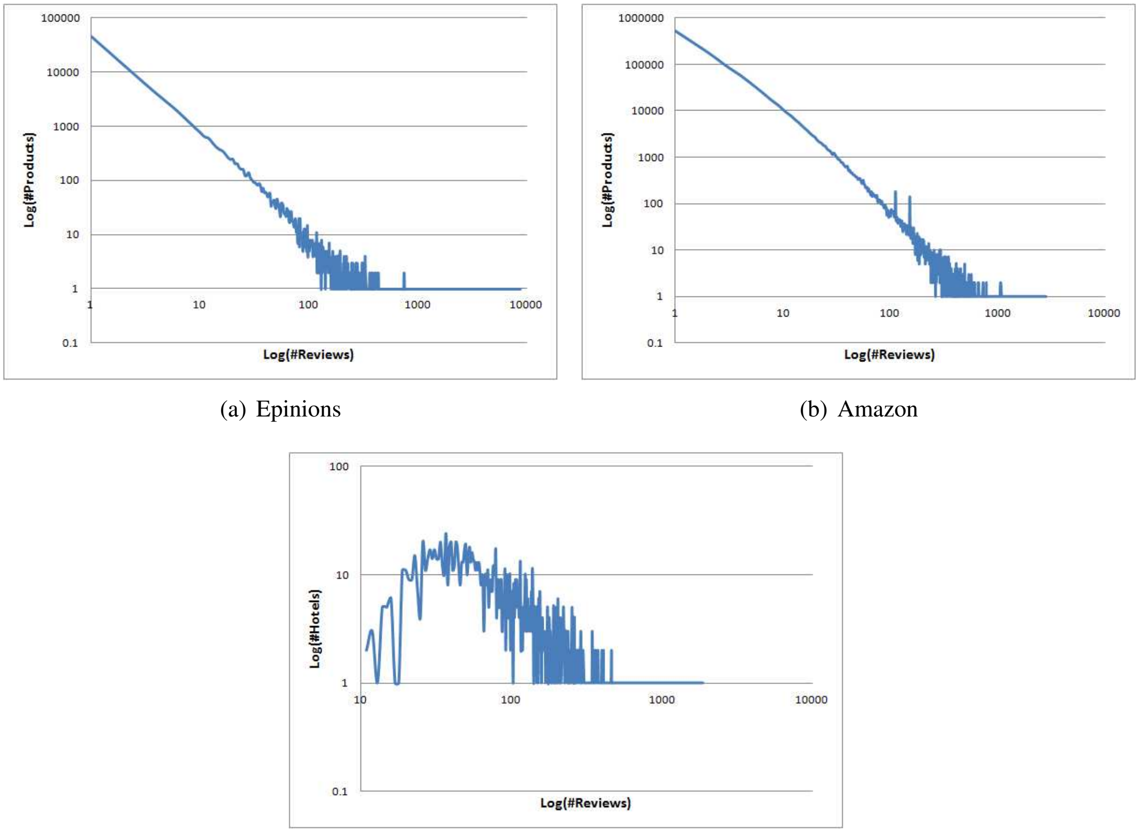
Regarding item categories, we used the available categorization in each dataset which were mostly at a high level (5 hotel categories based on their number of stars for TripAdvisor, 38 general categories for Amazon, and 379 product categories for Epinions). Table 7.2 shows some sample categories for each dataset.

Table 7.2: Sample categories of each dataset

Dataset	Sample Categories
Epinions	Accessories, Blazers, Dresses, Outerwear, Pants, Shirts, Skirts, ...
Amazon	Apparel, Electronics, Computers, Baby, Digital Games, Sports, Grocery, ...
TripAdvisor	1-star, 2-star, 3-star, 4-star, 5-star

All of the current works report only the average number of reviews per item, somehow masking the large percentage of cold start items in real-life datasets. In fact, cold start items are normally ignored in learning latent variable models. In order to show the variance in the numbers of reviews, Figures 7.4 and 7.5 show the distributions of #reviews per item and #reviews per reviewer in different datasets, respectively.

Not surprisingly, in the Epinions and Amazon datasets both distributions follow a power law. We



(a) Epinions

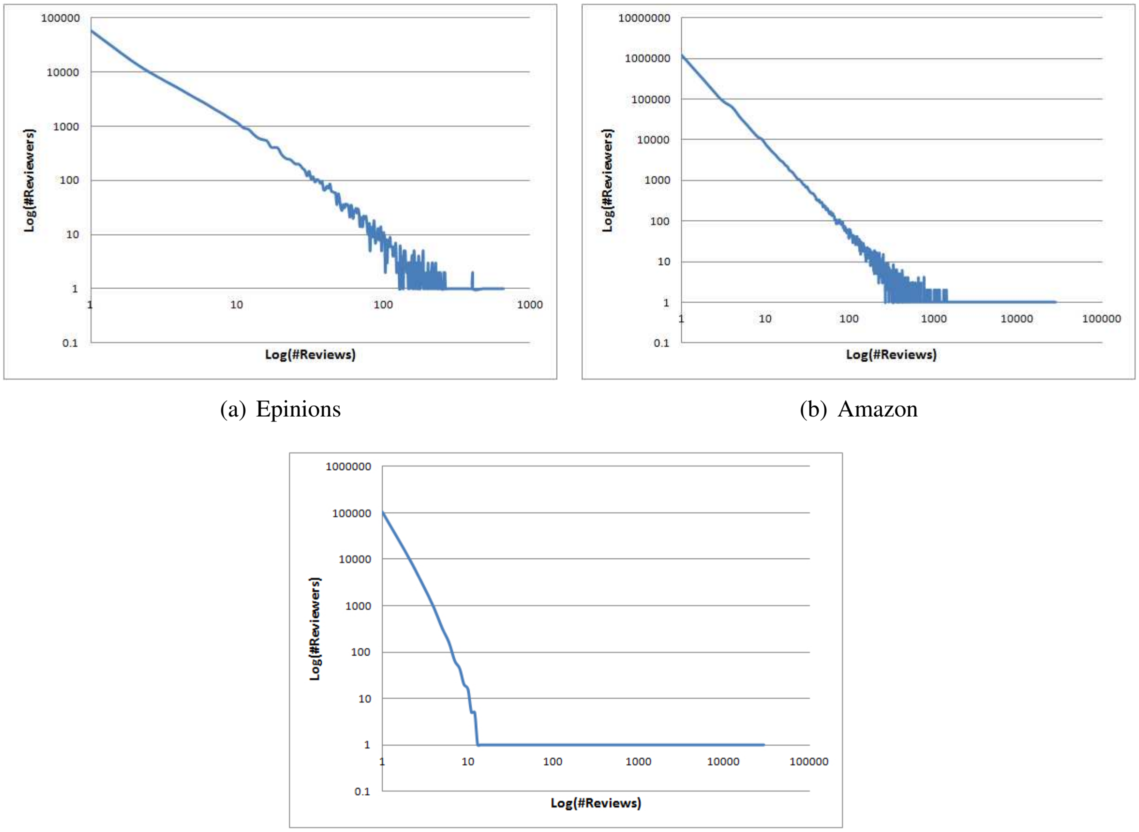
(b) Amazon

(c) TripAdvisor

Figure 7.3: Log-log plot of #reviews vs. #items

can see that a large number of items has only a few reviews, and a few items have a large number of reviews (Figures 7.3(a) and 7.3(b)). A similar property can be seen in the log-log plot of the number of reviews vs. the number of reviewers (Figures 7.4(a) and 7.4(b)). In the TripAdvisor dataset, the distribution of the number of reviews per reviewer (Figures 7.4(c)) also follows a power law. However, the relationship between the number of reviews and the number of hotels (Figure 7.3(c)) is below an ideal straight line for the first few points, since there are surprisingly few hotels with fewer than 50 reviews.

These power law distributions point out substantial diversity among items in real-life review datasets. To analyze the performance of the comparison partners separately on different types of items, we categorize items of each dataset into 5 groups based on the number of reviews. Table 7.3 shows the percentage of items in each dataset with the specified number of reviews.



(a) Epinions

(b) Amazon

(c) TripAdvisor

Figure 7.4: Log-log plot of #reviews vs. #reviewers

In the Epinions and Amazon datasets, more than 90% of products have less than 10 reviews which are considered cold start items. The TripAdvisor dataset has larger numbers of reviews per item. However, as Table 7.3 shows 31% of hotels have been reviewed by less than 50 reviewers which can be considered cold start in this dataset. These statistics clearly indicate that there is a need for opinion mining models with the focus on cold start items.

Table 7.4 also presents the average number of reviews per item for the defined item groups. It suggests that the average numbers of reviews for cold start items are indeed very small (2 for Epinions and Amazon, 25 for TripAdvisor) which makes it hard to learn an accurate model for these items.

Table 7.3: Percentage of items in each item group

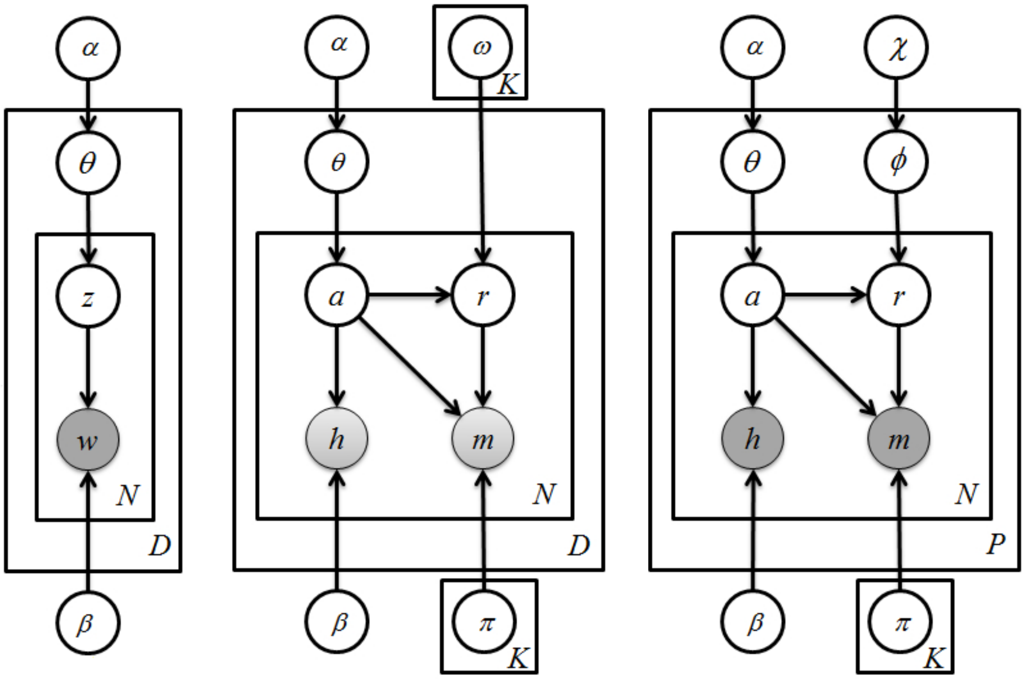
Item Groups	Epinions	Amazon	TripAdvisor
$1 < \#Rev \leq 10$	90%	91%	0%
$10 < \#Rev \leq 50$	8%	7%	31%
$50 < \#Rev \leq 100$	1%	1%	30%
$100 < \#Rev \leq 200$	$< 1\%$	$< 1\%$	24%
$200 < \#Rev$	$< 1\%$	$< 1\%$	14%

Table 7.4: Average #reviews per item in each item group

Item Groups	Epinions	Amazon	TripAdvisor
$1 < \#Rev \leq 10$	2	2	0
$10 < \#Rev \leq 50$	16	18	25
$50 < \#Rev \leq 100$	53	62	54
$100 < \#Rev \leq 200$	114	122	107
$200 < \#Rev$	324	338	297

7.4.2 Comparison Partners

We compare FLDA with the basic LDA model that generates all words of reviews [14] (Figure 7.5(a)) and the D-PLDA (ILDA) model presented in our previous work (Chapter 6) (Figure 7.5(b)). We selected these two models as comparison partners since experimental evaluation in Chapter 6 showed that the basic LDA performs best for cold start items and D-PLDA outperforms other models for non cold start items. Note that, FLDA adopts the same model for generating opinion phrases as D-PLDA, i.e., they have the same inner plate in the graphical model. Both LDA and D-PLDA are trained at the item level and D is the number of reviews for the given item. To tease apart the impact of the two major changes between D-PLDA and FLDA, we also compare a simplified version of FLDA, called I-FLDA, that does not model reviewers and their parameters but is trained at the category level (Figure 7.5(c)).



(a) LDA

(b) D-PLDA (ILDA)

(c) I-FLDA

Figure 7.5: Comparison partners: LDA and D-PLDA (ILDA) are state-of-the-art models, I-FLDA is a simplified version of FLDA

7.4.3 Experimental Evaluation

In this section, we evaluate the generalization performance of all comparison partners based on the likelihood of a held-out test set, which is standard in the absence of ground truth. For comparison, we trained all the latent variable models using EM with exactly the same stopping criteria and for various numbers of aspects, $k = \{5, 10, 15, 20, 25\}$. Since the relative results are similar for different values of k, we choose $k = 15$ for our discussion.

In the performance comparison, the goal is achieving high likelihood on a held-out test set. We hold out 10% of the reviews for testing purposes and use the remaining 90% to train models. As is standard for LDA models, we computed the perplexity of the held-out test set. A strong correlation of the perplexity and the accuracy (which can be computed only if ground truth is available) of aspect-based opinion mining models is shown in the previous chapter. The perplexity is monotonically decreasing in the likelihood of the test data, and a lower perplexity score indicates better performance.

Table 7.5: Perplexity comparison of different item groups in different datasets

(a) Epinions

Item Groups	LDA	D-PLDA	I-FLDA	FLDA
$1 < \#Rev \leq 10$	4413.65	5413.65	4187.98	3287.98
$10 < \#Rev \leq 50$	2338.67	1975.34	1903.45	1687.67
$50 < \#Rev \leq 100$	1671.23	592.39	588.61	468.12
$100 < \#Rev \leq 200$	1394.72	164.18	153.02	133.90
$200 < \#Rev$	1385.99	142.37	142.16	140.35

(b) Amazon

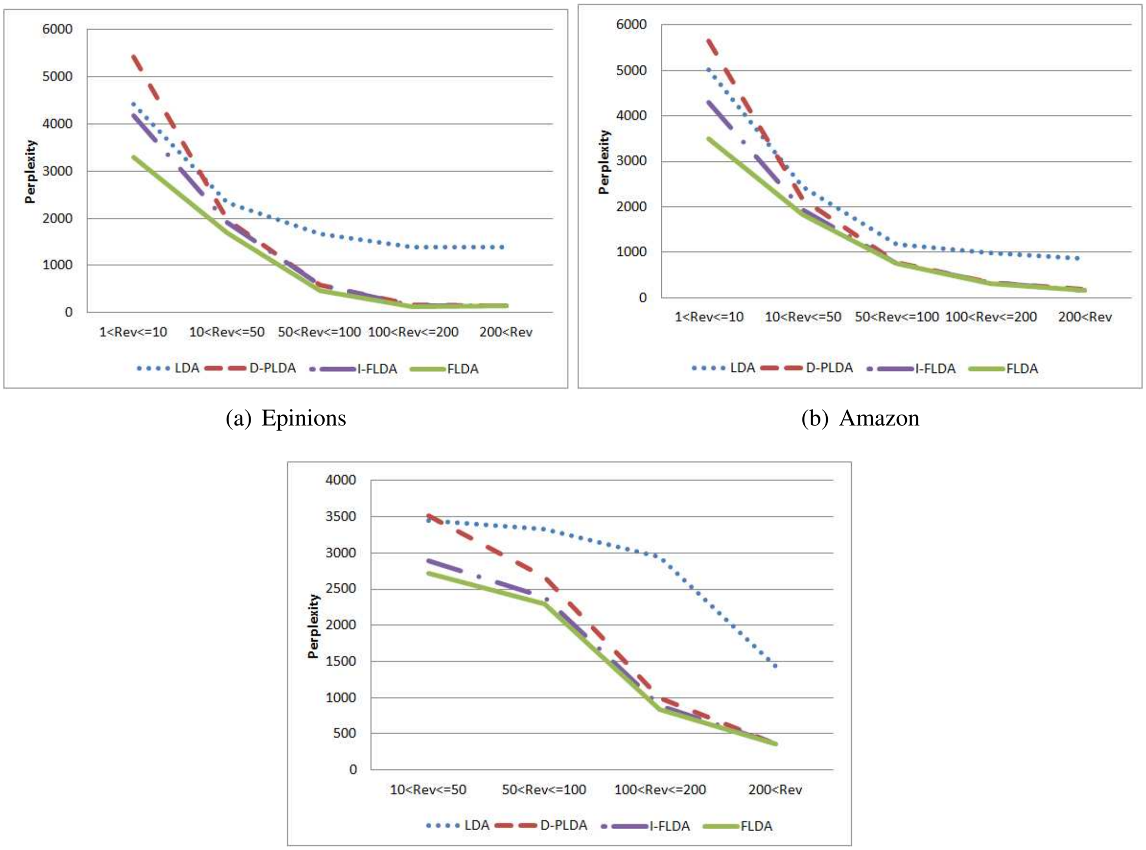
Item Groups	LDA	D-PLDA	I-FLDA	FLDA
$1 < \#Rev \leq 10$	5019.79	5653.79	4302.45	3494.01
$10 < \#Rev \leq 50$	2434.71	2159.02	1931.34	1833.66
$50 < \#Rev \leq 100$	1183.14	769.77	756.09	744.18
$100 < \#Rev \leq 200$	993.78	339.69	331.45	318.49
$200 < \#Rev$	869.25	177.08	173.15	172.45

(c) TripAdvisor

Item Groups	LDA	D-PLDA	I-FLDA	FLDA
$1 < \#Rev \leq 10$	-	-	-	
$10 < \#Rev \leq 50$	3446.61	3518.95	2898.56	2725.76
$50 < \#Rev \leq 100$	3336.61	2673.19	2394.09	2301.31
$100 < \#Rev \leq 200$	2943.46	1003.09	892.59	843.91
$200 < \#Rev$	1438.59	363.20	362.74	359.03

Table 7.5 and Figure 7.6 present the perplexity results of FLDA and the comparison partners for different groups of items in different datasets. The first observation, that has already been discussed in our previous work (Chapter 6), is that the D-PLDA model outperforms LDA in all datasets for non cold start items. However, for cold start items it has higher perplexity than LDA, indicating poor performance of the model in the absence of enough training data. We can also observe that I-FLDA, which is trained at the category level but does not model reviewers, achieves lower perplexity than D-PLDA, especially for cold start items. This better performance can be explained by the fact that I-FLDA is trained at the category level and learns the latent factors using the reviews of all the items of a category, in particular the non cold start items, and uses them as prior for cold start items.

Finally, we note that in all datasets and for all item groups, FLDA consistently outperforms LDA,



(a) Epinions

(b) Amazon

(c) TripAdvisor

Figure 7.6: Perplexity results of all comparison partners for different datasets

D-PLDA and I-FLDA. These findings show that FLDA's assumptions regarding using the category level information for aspect extraction and the user modeling for rating prediction are appropriate. The perplexity gain of FLDA is most notable for cold start items underlining the effectiveness of FLDA in modeling such items. For items with large numbers of reviews, FLDA can slightly improve the performance of I-FLDA by also modeling reviewers. Comparing the results of FLDA, I-FLDA and D-PLDA shows that when there is enough training data (reviews), learning a model at the item level is promising.

7.5 Applications

In the following sections we perform two application-oriented evaluations to demonstrate the gains of FLDA in practice.

7.5.1 Item Categorization

One of the applications of category-level models is the ability of categorizing new items based on their reviews, e.g., identifying the class of a hotel (1 to 5 star), or type of a book (e.g., children's books, textbooks, audio books, magazines, etc.) based on their reviews. This feature is especially beneficial when working with uncategorized reviews, e.g., forums, Blogs, discussion groups, etc.

In [14], Blei et al. proposed to use the basic LDA model for document classification. In particular, LDA is used as a dimensionality reduction method, as it reduces any document to a vector of real-valued features, i.e., the posterior Dirichlet parameter associated with each document. The parameters of an LDA model are learned using all the documents, without reference to their true class label. The topic distribution provides a low-dimensional representation (feature vector) of a document, and a Support Vector Machine (SVM) is trained on these feature vectors to distinguish the classes.

In our scenario, we can adopt the same approach for item categorization. The FLDA model can be used to produce feature vectors for item categorization as follows. We first estimate the parameters of the FLDA model using all the reviews of all items of all categories. The learned topic distribution θ of an item is used as the feature vector of that item, and an SVM classifier is trained on these feature vectors to classify items into categories (FLDA-SVM). Note that, the topic distribution of an item in this model cannot be interpreted as the aspect distribution of the item.

Since the LDA and D-PLDA models learn topic distributions of reviews, not items, they cannot be directly used as comparison partners for item categorization. However, by applying these models at the category level, we can obtain the topic distribution of items as item feature vectors. These models use all reviews of all items of all categories to learn the feature vectors of items (similar to FLDA-SVM). As a baseline, we also train a classifier on simple bag-of-words features (BOW-SVM). Table 7.6 shows the accuracy of SVM classifiers for cold start and non cold start items trained on different feature spaces.

The first observation is that for all feature sets the accuracy of item categorization is higher for non cold start items than for cold start items. This was predictable as there is more training data for non cold start items. Comparing BOW-SVM with LDA-SVM and D-PLDA-SVM, we can see an

Table 7.6: Average accuracy of SVM classifiers trained on different feature sets for item categorization

Dataset	Epinions		Amazon		TripAdvisor	
Item Type	cold	non	cold	non	cold	non
BOW-SVM	64%	88%	62%	88%	68%	91%
LDA-SVM	71%	90%	67%	91%	73%	93%
D-PLDA-SVM	79%	96%	75%	94%	85%	97%
FLDA-SVM	83%	96%	79%	95%	86%	97%

increase in classification accuracy by using the LDA-based features. This suggests that the topic-based representation provided by LDA can be useful for feature selection in item categorization. We also observe that the classification accuracy is substantially improved by using FLDA features. This suggests that the learned item factors of FLDA can provide a more accurate low-dimensional feature set for item categorization. Note that the LDA-based models ($k = 15$) reduce the feature space of the Epinions, Amazon, and TripAdvisor datasets by 97%, 99%, and 92%, respectively compared to all word features.

7.5.2 Overall Rating Prediction for Reviews

In most of the reviewing websites, reviewers are asked to assign an overall rating (as a number in some given range) to express their overall level of satisfaction with the reviewed item. However, in other repositories of reviews, such as forums and Blogs, such overall ratings are not normally provided. One of the applications of FLDA is the ability of predicting the overall rating of a review. As each review is written by a reviewer about an item, the overall rating of a review depends on both item and reviewer factors. The aspect and rating distributions of items and reviewers learned by the FLDA model can be used for computing the overall rating of the review as follows.

In recommender systems, Matrix Factorization (MF) is employed to factorize the $user \times item$ rating matrix to predict the rating of a user for an item [144, 73, 74]. Inspired by this model, we can compute the overall rating of an item by a reviewer using the learned item and reviewer factors. In the FLDA model, the latent aspect distribution of review d_{pu} is determined by the aspect distributions of the corresponding item, p, and reviewer, u, and is denoted by $P(a|\theta_p, \vartheta_u)$. In the same way the latent rating distribution of review d_{pu} is denoted by $P(r|a, \phi_p, \varphi_u)$. According to the probabilistic MF model, the distribution of the overall ratings o_{up} for user u and item p, can be computed as follows:

$$P(o_{up} = r) = \sum_{a} P(a|\theta_p, \vartheta_u) P(r|a, \phi_p, \varphi_u) \tag{7.4}$$

Since in the review datasets we used, ratings are chosen from the set $\{1, 2, 3, 4, 5\}$, we define 5 classes of overall ratings. For each item we train an SVM classifier on the distribution of the overall ratings acquired by Equation 7.4 to classify the overall rating of a given review (FLDA-SVM). As comparison partners, we train two classifiers on the review feature vectors generated by LDA (LDA-SVM) and D-PLDA (D-PLDA-SVM). The review feature vector of LDA is the topic distribution of the review, and the review feature vector of D-PLDA is the distribution of the overall ratings obtained using the probabilistic MF model (similar to Equation 7.4). We also train a classifier on simple bag-of-words features (BOW-SVM) as a baseline. Table 7.7 shows the accuracy of SVM classifiers for cold start and non cold items trained on different feature spaces.

Table 7.7: Average accuracy of SVM classifiers trained on different feature sets for overall rating prediction

Dataset	Epinions		Amazon		TripAdvisor	
Item Type	cold	non	cold	non	cold	non
BOW-SVM	49%	83%	44%	79%	47%	82%
LDA-SVM	56%	85%	53%	80%	59%	85%
D-PLDA-SVM	57%	86%	54%	80%	63%	87%
FLDA-SVM	72%	89%	70%	83%	74%	91%

Again we can see that for all feature sets the accuracy of overall rating prediction for non cold start items is much higher than that of cold start items, and also the accuracy of all LDA-based models is higher than bag-of-words features. The accuracy of D-PLDA-SVM is slightly higher than that of LDA-SVM as it uses the rating distribution of the item for generating the feature vectors of reviews. Finally, as shown in Table 7.7, the accuracy of FLDA-SVM for the task of overall rating prediction is much higher than that of the comparison partners. This suggests that for a given review the learned item and user factors can be used as a low-dimensional feature set for predicting its overall rating.

7.6 Conclusion

Aspect-based opinion mining is the problem of automatically extracting aspects and estimating their ratings from reviews. All of the current models are trained at the item level (a model is trained form all reviews of an item) to perform these tasks. In this paper, we argued that while learning a model at the item level is fine for frequently reviewed items, it is ineffective for items with few reviews (cold start items). Note that, more than 90% of products in Epinions and Amazon datasets and 30% of hotels in the TripAdvisor dataset are cold start.

Addressing this need, we introduced the problem of aspect-based opinion mining for cold start items and proposed a probabilistic model based on LDA, called FLDA. Our model assumes that aspects in a review are sampled from the aspect distributions of the corresponding item and reviewer and the rating of each aspect is sampled conditioned on that aspect and the rating distributions of the item and reviewer. For cold start items the aspect distribution is mainly determined by the prior aspect distribution of the category, and the rating distribution of each aspect is mainly determined by the rating distribution of the reviewer (or by the prior rating distribution of all reviewers if the reviewer is cold start). The aspect and rating distributions for non cold start items are mainly determined by the observed reviews of that item.

We conducted extensive experiments on three real-life datasets and compared FLDA against the baseline LDA, the state-of-the-art D-PLDA, and the simplified I-FLDA models. FLDA clearly outperforms all of the comparison partners in terms of likelihood of the test set. For cold start items, the perplexity gain of FLDA is very large. We argued that the major reason for this gain is using the category level information and also modeling reviewers. We further teased apart the impact of modeling reviewers by comparing FLDA with the simplified I-FLDA model showing that modeling reviewers significantly impacts the model performance for cold start items. We also demonstrated the accuracy of FLDA in two applications: categorizing items and predicting the overall rating of reviews based on the learned feature vectors.

This work suggests several directions for future research. FLDA assumes a given definition of item categories, but there may be alternative options to define them. For example, is it better to categorize hotels based on stars, or location, or price? An item taxonomy is a hierarchical structure of categories and subcategories. For example, the hierarchy for the category 'MP3 Players' could be "Electronics $>$ Audio $>$ Audio Players & Recorders $>$ MP3 Players". In addition, in a scenario with a given item taxonomy, it would be interesting to explore methods to automatically learn the granularity (taxonomy) level that leads to the best model performance.

Chapter 8

Conclusion

Opinion mining has become a fascinating research area due to the availability of a huge volume of user-generated content, e.g., reviewing websites, forums, and blogs. Aspect-based opinion mining, which aims to extract item aspects and their corresponding ratings from online reviews, is a relatively new sub-area that attracted a great deal of attention recently. In this thesis, we focused on this problem because of its key role in the area of opinion mining. The extracted aspects and estimated ratings not only ease the process of decision making for customers but also can be utilized in other opinion mining systems. In Chapter 3, we defined this problem formally and reviewed the state-of-the-art approaches presented in the literature.

In Chapter 4, we introduced a hybrid method, called Opinion Digger [109], for the considered problem. Opinion Digger takes advantages of both frequency- and relation-based approaches to identify aspects and estimate their rating. Opinion Digger finds the aspect-sentiment relations by mining a set of opinion patterns from reviews. Then it uses the mined pattern to filter out non-aspects from frequent noun phrases. It also uses a novel technique for grouping synonymous aspects. Regarding rating prediction, while previous works just determine whether people's opinion about an aspect is positive or negative, Opinion Digger precisely determines the strength of positiveness or negativeness of an opinion by estimating a rating in the range [1,5]. Evaluation of results showed that combining the idea of frequency and relation-based approaches can effectively improve the accuracy of aspect extraction.

The need for manual tuning of various parameters makes the frequency- and relation-based methods hard to port to another dataset. As a result, in our next work we moved on to model-based approaches and proposed a probabilistic graphical model based on LDA, called ILDA [111], to automatically learn the model parameters from the data. In comparison to the previous works which perform aspect identification and rating prediction in separate steps (leading to the accumulation of errors), ILDA jointly identify aspects and predict their ratings from online reviews. The details of ILDA and the experimental evaluation on a real life data set from the Epinions.com are presented in Chapter 5. ILDA outperforms all of the comparison partners in terms of likelihood of the test set, and accuracy of aspect and rating clusters. We argued that the major reason for the consistent enhancement is that the underlying probabilistic assumptions of ILDA (interdependency between aspects and ratings) are appropriate for the problem domain

In Chapter 6, we compared ILDA with a series of increasingly sophisticated LDA-based models representing the essence of the major published methods. This comprehensive comparison allowed us to tease apart the impact of various design decisions and drove some guidelines for designing future models. We evaluated the performance of these models in terms of the likelihood of a held-out test set on a very large real-life data set from Epinions.com. As a novel technical contribution, we presented a method for preprocessing reviews based on grammatical relations provided by a dependency parser. Comparison of this technique with others showed that the generated opinion phrases are more accurate than the current preprocessing techniques. We also measured the impact of the size of the training data and performed experiments for different subsets of items with different numbers of reviews. The results indicated that while ILDA works best for items with large number of reviews, it performs poorly when the size of the training dataset is small, i.e., for cold start items.

The cold start problem is critical in real-life data sets, such as those from Epinions.com and Amazon.com, in which more than 90% of items are cold start. We addressed this problem in Chapter 7 and proposed a probabilistic graphical model based on LDA, called FLDA. FLDA models items and reviewers by a set of latent factors and learns them using reviews of an item category. For cold start items the aspect distribution is mainly determined by the prior aspect distribution of the category, and the rating distribution of each aspect is mainly determined by the rating distribution of the reviewer. The aspect and rating distributions for non cold start items are mainly determined by the observed reviews of that item. Experimental results on real life data sets from Epinions.com, Amazon.com, and TripAdvisor.com showed that our proposed model achieve significant quality gain for cold start items compared to the state-of-the-art models.

8.1 Future Research Directions

The research of this thesis suggests many promising directions for future research in the field of aspect-based opinion mining. In this section, we briefly discuss such directions:

Directions in the area of natural language processing:

- **Implicit aspects**:

  Most of the current works extract only explicit aspects. However, there are usually many types of implicit aspect expressions in a review. Adjectives and adverbs are perhaps the most common types because most adjectives describe some specific attributes or properties of entities, e.g., expensive describes 'price,' and beautiful describes 'appearance.' Implicit aspects can be verbs too. In general, implicit aspect expressions can be very complex, e.g., 'This camera will not easily fit in a pocket.' 'fit in a pocket' indicates the aspect size. Although there have been some works considering extraction of implicit aspects, further research is still needed.

- **More complex sentiments**:

  Most sentiments are expressed through adjectives and adverbs. However, nouns (e.g., rubbish, junk, and crap) and verbs (e.g., hate and love) can also be used to express sentiments. Apart from individual words, there are also sentiment phrases and idioms, e.g., cost someone an arm and a leg. While identifying these types of sentiments are very difficult, the main challenge is predicting the polarity/rating of them.

- **Co-reference resolution**:

  Co-reference resolution has been studied extensively in the NLP community but not in the context of opinion mining. If we do not resolve co-references in opinion sentence, but only consider opinions in each sentence in isolation, we will lose recall.

- **Comparative sentences**:

  Sometimes opinions are expressed in sentences comparing two items. Two main problems to be addressed are 1) identifying comparative sentences and 2) determining the preferred item. Although there have been some existing works, further research is still needed.

- **Extraction of opinion phrases**:

  As discussed in section 4.7, bag-of-opinion phrases models can outperform bag-of-words topic models and using the semantic relationship between words pays off for extracting opinion phrases. More sophisticated methods for extracting opinion phrases are needed to further increase the accuracy of aspect-based opinion mining.

- **Other types of opinionated documents**:

  Researchers working on aspect-based opinion mining have focused mainly on online reviews. These forms of data are relatively easy to handle because reviews are opinion rich and have little irrelevant information. However, other forms of opinion text such as forum discussions and commentaries are much harder to deal with because they are mixed with all kinds of non-opinion contents and often talk about multiple items and involve user interactions. Identifying techniques for dealing with noisy information can be a promising research direction.

Directions in the area of data mining and machine learning:

- **Additional input data**:

  The impact of different additional input sources in topic modeling approach (e.g. review's overall rating, known aspects, rating guideline) should be explored. While some of the current works used the additional information in their models, a comprehensive performance comparison is needed to clarify the impact of each additional input source in improving the performance.

- **Evaluation**:

  One of the issues in topic modeling methods in general, and in particular in opinion mining, is the evaluation. As discussed in Section 3.4, obtaining a ground truth to evaluate the accuracy of a method is normally expensive as it typically requires manual labeling. On the other hand, evaluation of topic modeling methods based on the likelihood of a held-out test set is applicable to data sets without such ground truth. However, this evaluation does not normally clarify the effectiveness of the model in terms of the accuracy of the extracted aspects and estimated ratings. Further research is needed in this area to find more application-oriented evaluation metrics.

Bibliography

[1] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, pages 487–499, San Francisco, CA, USA, 1994. Morgan Kaufmann Publishers Inc.

[2] Nikolay Archak, Anindya Ghose, and Panagiotis G. Ipeirotis. Show me the money!: deriving the pricing power of product features by mining consumer reviews. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '07, pages 56–65, New York, NY, USA, 2007. ACM.

[3] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Multi-facet rating of product reviews. In *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, ECIR '09, pages 461–472, Berlin, Heidelberg, 2009. Springer-Verlag.

[4] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC '10)*, 2010.

[5] Anton Bakalov, Ariel Fuxman, Partha Pratim Talukdar, and Soumen Chakrabarti. Scad: collective discovery of attribute values. In *Proceedings of the 20th international conference on World wide web*, WWW '11, pages 447–456, New York, NY, USA, 2011. ACM.

[6] Alexandra Balahur, Ester Boldrini, Andrés Montoyo, and Patricio Martínez-Barco. Going beyond traditional qa systems: Challenges and keys in opinion question answering. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING '10)*, pages 27–35, 2010.

[7] Alexandra Balahur, Ester Boldrini, Andrés Montoyo, and Patricio Martínez-Barco. Opinion question answering: Towards a unified approach. In *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI '10)*, pages 511–516, 2010.

[8] Carmen Banea, Rada Mihalcea, Janyce Wiebe, and Samer Hassan. Multilingual subjectivity analysis using machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 127–135, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[9] Shenghua Bao, Shengliang Xu, Li Zhang, Rong Yan, Zhong Su, Dingyi Han, and Yong Yu. Joint emotion-topic modeling for social affective text mining. In *Proceedings of the 2009 Ninth IEEE International Conference on Data Mining*, ICDM '09, pages 699–704, Washington, DC, USA, 2009. IEEE Computer Society.

[10] Steven Bethard, Hong Yu, Ashley Thornton, Vasileios Hatzivassiloglou, and Dan Jurafsky. Automatic extraction of opinion propositions and their holders. In *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text*, pages 22–24, 2004.

[11] Sasha Blair-goldensohn, Tyler Neylon, Kerry Hannan, George A. Reis, Ryan Mcdonald, and Jeff Reynar. Building a sentiment summarizer for local service reviews. In *Proceedings of the international conference on World Wide Web workshop on NLP in the Information Explosion Era (WWW workshop '08),*, 2008.

[12] David M. Blei and Michael I. Jordan. Modeling annotated data. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, SIGIR '03, pages 127–134, New York, NY, USA, 2003. ACM.

[13] David M. Blei and John D. Lafferty. Dynamic topic models, 2006.

[14] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.

[15] Erik Boiy and Marie-Francine Moens. A machine learning approach to sentiment analysis in multilingual web texts. *Inf. Retr.*, 12(5):526–558, 2009.

[16] Johan Bos and Malvina Nissim. An empirical approach to the interpretation of superlatives. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 9–17, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[17] S. R. K. Branavan, Harr Chen, Jacob Eisenstein, and Regina Barzilay. Learning document-level semantic properties from free-text annotations. *J. Artif. Int. Res.*, 34, 2009.

[18] Clodagh Brien. The emergence of the social media empowered consumer. *Irish Marketing Review*, 2011.

[19] Samuel Brody and Noemie Elhadad. An unsupervised aspect-sentiment model for online reviews. In *Proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 804–812, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[20] Jaime Carbonell and Jade Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pages 335–336, New York, NY, USA, 1998. ACM.

[21] Marie catherine De Marneffe, Bill Maccartney, and Christopher D. Manning. Generating typed dependency parses from phrase structure parses. In *Proceedings of the international conference on Language Resources and Evaluation (LREC*, pages 449–454, 2006.

[22] Yejin Choi and Claire Cardie. Hierarchical sequential learning for extracting opinions and their attributes. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL '10)*, pages 269–274, 2010.

[23] Koby Crammer and Yoram Singer. Pranking with ranking. In *Proceedings of the Advances in Neural Information Processing Systems 14*, pages 641–647. MIT Press, 2001.

[24] Cristian Danescu-Niculescu-Mizil, Gueorgi Kossinets, Jon M. Kleinberg, and Lillian Lee. How opinions are received by online communities: a case study on amazon.com helpfulness votes. In *Proceedings of the 18th International Conference on World Wide Web (WWW '09)*, pages 141–150, 2009.

[25] Sanjiv R. Das, Mike Y. Chen, To Vikas Agarwal, Chris Brooks, Yuk shee Chan, David Gibson, David Leinweber, Asis Martinez-jerez, Priya Raghubir, Sridhar Rajagopalan, Ajit Ranade, Mark Rubinstein, and Peter Tufano. Yahoo! for amazon: Sentiment extraction from small talk on the web. 2001.

[26] Kushal Dave, Steve Lawrence, and David M. Pennock. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web*, WWW '03, pages 519–528, New York, NY, USA, 2003. ACM.

[27] Jorge Carrillo de Albornoz, Laura Plaza, Pablo Gervás, and Alberto Díaz. A joint model of feature mining and sentiment analysis for product review rating. In *Proceedings of the 33rd European conference on Advances in information retrieval*, ECIR '11, pages 55–66, Berlin, Heidelberg, 2011. Springer-Verlag.

[28] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, series* B, 39(1):1–38, 1977.

[29] Xiaowen Ding and Bing Liu. Resolving object and attribute coreference in opinion mining. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 268–276, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[30] Xiaowen Ding, Bing Liu, and Philip S. Yu. A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, WSDM '08, pages 231–240, New York, NY, USA, 2008. ACM.

[31] Anlei Dong, Yi Chang, Zhaohui Zheng, Gilad Mishne, Jing Bai, Ruiqiang Zhang, Karolina Buchner, Ciya Liao, and Fernando Diaz. Towards recency ranking in web search. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 11–20, New York, NY, USA, 2010. ACM.

[32] Weifu Du and Songbo Tan. An iterative reinforcement approach for fine-grained opinion mining. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, pages 486–493, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[33] Koji Eguchi and Victor Lavrenko. Sentiment retrieval using generative models. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 345–354, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[34] Elena Erosheva, Stephen Fienberg, and John Lafferty. Mixed-membership models of scientific publications. volume 101, 2004.

[35] Christiane Fellbaum. Wordnet: An electronic lexical database. *Cambridge, MA: MIT Press*, 1998.

[36] Marcelo Fiszman, Dina Demner-Fushman, Francois M. Lang, Philip Goetz, and Thomas C. Rindflesch. Interpreting comparative constructions in biomedical text. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*, BioNLP '07, pages 137–144, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.

[37] Osamu Furuse, Nobuaki Hiroshima, Setsuo Yamada, and Ryoji Kataoka. Opinion sentence search engine on open-domain blog. In *Proceedings of the 20th international joint conference on Artifical intelligence*, IJCAI'07, pages 2760–2765, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[38] Murthy Ganapathibhotla and Bing Liu. Mining opinions in comparative sentences. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 241–248, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[39] Shima Gerani, Mark James Carman, and Fabio Crestani. Proximity-based opinion retrieval. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 403–410, New York, NY, USA, 2010. ACM.

[40] Anindya Ghose and Panagiotis G. Ipeirotis. Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics. *IEEE Trans. Knowl. Data Eng.*, 23(10):1498–1512, 2011.

[41] Honglei Guo, Huijia Zhu, Zhili Guo, and Zhong Su. Domain customization for aspect-oriented opinion analysis with multi-level latent sentiment clues. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pages 2493–2496, New York, NY, USA, 2011. ACM.

[42] Honglei Guo, Huijia Zhu, Zhili Guo, XiaoXun Zhang, and Zhong Su. Product feature categorization with multilevel latent semantic association. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 1087–1096, New York, NY, USA, 2009. ACM.

[43] Zhen Hai, Kuiyu Chang, and Jung-jae Kim. Implicit feature identification via co-occurrence association rule mining. In *Proceedings of the 12th international conference on Computational linguistics and intelligent text processing - Volume Part I*, CICLing'11, pages 393–404, Berlin, Heidelberg, 2011. Springer-Verlag.

[44] Ahmed Hassan and Dragomir Radev. Identifying text polarity using random walks. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 395–403, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[45] Vasileios Hatzivassiloglou and Kathleen R. McKeown. Predicting the semantic orientation of adjectives. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, EACL '97, pages 174–181, Stroudsburg, PA, USA, 1997. Association for Computational Linguistics.

[46] Yulan He, Chenghua Lin, and Harith Alani. Automatically extracting polarity-bearing topics for cross-domain sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 123–131, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[47] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 50–57, 1999.

[48] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04, pages 168–177, New York, NY, USA, 2004. ACM.

[49] Minqing Hu and Bing Liu. Mining opinion features in customer reviews. In *Proceedings of the 19th national conference on Artifical intelligence*, AAAI'04, pages 755–760. AAAI Press, 2004.

[50] Minqing Hu and Bing Liu. Opinion extraction and summarization on the web. In *Proceedings of the 21st national conference on Artificial intelligence - Volume 2*, AAAI'06, pages 1621–1624. AAAI Press, 2006.

[51] Shen Huang, Dan Shen, Wei Feng, Catherine Baudin, and Yongzheng Zhang. Improving product review search experiences on general search engines. In *Proceedings of the 11th International Conference on Electronic Commerce*, ICEC '09, pages 107–116, New York, NY, USA, 2009. ACM.

[52] Shen Huang, Dan Shen, Wei Feng, Yongzheng Zhang, and Catherine Baudin. Discovering clues for review quality from author's behaviors on e-commerce sites. In *Proceedings of the 11th International Conference on Electronic Commerce*, ICEC '09, pages 133–141, New York, NY, USA, 2009. ACM.

[53] Xuanjing Huang and W. Bruce Croft. A unified relevance model for opinion retrieval. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 947–956, New York, NY, USA, 2009. ACM.

[54] Niklas Jakob and Iryna Gurevych. Using anaphora resolution to improve opinion target identification in movie reviews. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, pages 263–268, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[55] Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. Target-dependent twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 151–160, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[56] Peng Jiang, Chunxia Zhang, Hongping Fu, Zhendong Niu, and Qing Yang. An approach based on tree kernels for opinion mining of online product reviews. In *Proceedings of the 10th IEEE International Conference on Data Mining (ICDM '10)*, pages 256–265, 2010.

[57] Valentin Jijkoun, Maarten de Rijke, and Wouter Weerkamp. Generating focused topic-specific sentiment lexicons. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 585–594, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[58] Wei Jin, Hung Hay Ho, and Rohini K. Srihari. Opinionminer: a novel machine learning system for web opinion mining and extraction. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 1195–1204, New York, NY, USA, 2009. ACM.

[59] Nitin Jindal and Bing Liu. Identifying comparative sentences in text documents. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '06)*, pages 244–251, 2006.

[60] Nitin Jindal and Bing Liu. Mining comparative sentences and relations. In *Proceedings of the 21st national conference on Artificial intelligence - Volume 2*, AAAI '06, pages 1331–1336. AAAI Press, 2006.

[61] Nitin Jindal and Bing Liu. Review spam detection. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 1189–1190, New York, NY, USA, 2007. ACM.

[62] Nitin Jindal and Bing Liu. Opinion spam and analysis. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, WSDM '08, pages 219–230, New York, NY, USA, 2008. ACM.

[63] Nitin Jindal, Bing Liu, and Ee-Peng Lim. Finding unusual review patterns using unexpected rules. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1549–1552, New York, NY, USA, 2010. ACM.

[64] Yohan Jo and Alice H. Oh. Aspect and sentiment unification model for online review analysis. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 815–824, New York, NY, USA, 2011. ACM.

[65] Nobuhiro Kaji and Masaru Kitsuregawa. Building lexicon for sentiment analysis from massive collection of html documents. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL '07)*, pages 1075–1083, 2007.

[66] Hiroshi Kanayama and Tetsuya Nasukawa. Fully automatic lexicon expansion for domain-oriented sentiment analysis. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP '06)*, pages 355–363, 2006.

[67] Noriaki Kawamae. Predicting future reviews: sentiment analysis models for collaborative filtering. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 605–614, New York, NY, USA, 2011. ACM.

[68] Hyun Duk Kim and ChengXiang Zhai. Generating comparative summaries of contradictory opinions in text. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 385–394, New York, NY, USA, 2009. ACM.

[69] Jungi Kim, Jin-Ji Li, and Jong-Hyeok Lee. Evaluating multilanguage-comparability of subjectivity analysis systems. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 595–603, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[70] Soo-Min Kim and Eduard Hovy. Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[71] Soo-Min Kim, Patrick Pantel, Tim Chklovski, and Marco Pennacchiotti. Automatically assessing review helpfulness. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 423–430, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[72] Nozomi Kobayashi, Ryu Iida, Kentaro Inui, and Yuji Matsumoto. Opinion mining on the web by extracting subject-aspect-evaluation relations. In *Proceedings of the AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, pages 86–91, 2006.

[73] Yehuda Koren. Collaborative filtering with temporal dynamics. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 447–456, New York, NY, USA, 2009. ACM.

[74] Yehuda Koren, Robert M. Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42(8):30–37, 2009.

[75] Lun-Wei Ku, Yu-Ting Liang, and Hsin-Hsi Chen. Opinion extraction, summarization and tracking in news and blog corpora. In *Proceedings of the AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, pages 100–107, 2006.

[76] Lun-Wei Ku, Yu-Ting Liang, and Hsin-Hsi Chen. Question analysis and answer passage retrieval for opinion question answering systems. volume 13, 2008.

[77] Himabindu Lakkaraju, Chiranjib Bhattacharyya, Indrajit Bhattacharya, and Srujana Merugu. Exploiting coherence for the simultaneous discovery of latent facets and associated sentiments. In *Proceedings of the Eleventh SIAM International Conference on Data Mining (SDM)*, pages 498–509, 2011.

[78] Kevin Lerman and Ryan McDonald. Contrastive summarization: an experiment with consumer reviews. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, NAACL-Short '09, pages 113–116, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[79] Baoli Li, Yandong Liu, Ashwin Ram, Ernest V. Garcia, and Eugene Agichtein. Exploring question subjectivity prediction in community qa. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 735–736, New York, NY, USA, 2008. ACM.

[80] Beibei Li, Anindya Ghose, and Panagiotis G. Ipeirotis. Towards a theory model for product search. In *Proceedings of the 20th international conference on World wide web*, WWW '11, pages 327–336, New York, NY, USA, 2011. ACM.

[81] Fangtao Li, Chao Han, Minlie Huang, Xiaoyan Zhu, Ying-Ju Xia, Shu Zhang, and Hao Yu. Structure-aware review mining and summarization. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 653–661, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[82] Fangtao Li, Minlie Huang, and Xiaoyan Zhu. Sentiment analysis with global topics and local dependency. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*, 2010.

[83] Fangtao Li, Nathan Liu, Hongwei Jin, Kai Zhao, Qiang Yang, and Xiaoyan Zhu. Incorporating reviewer and product information for review rating prediction. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence - Volume Volume Three*, IJCAI'11, pages 1820–1825. AAAI Press, 2011.

[84] Fangtao Li, Yang Tang, Minlie Huang, and Xiaoyan Zhu. Answering opinion questions with random walks on graphs. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 737–745, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[85] Yung-Ming Li, Cheng-Yang Lai, and Ching-Wen Chen. Identifying bloggers with marketing influence in the blogosphere. In *Proceedings of the 11th International Conference on Electronic Commerce*, ICEC '09, pages 335–340, New York, NY, USA, 2009. ACM.

[86] Zhichao Li, Min Zhang, Shaoping Ma, Bo Zhou, and Yu Sun. Automatic extraction for product feature words from comments on the web. In *Proceedings of the 5th Asia Information Retrieval Symposium on Information Retrieval Technology*, AIRS '09, pages 112–123, Berlin, Heidelberg, 2009. Springer-Verlag.

[87] Ee-Peng Lim, Viet-An Nguyen, Nitin Jindal, Bing Liu, and Hady Wirawan Lauw. Detecting product review spammers using rating behaviors. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 939–948, New York, NY, USA, 2010. ACM.

[88] Chenghua Lin and Yulan He. Joint sentiment/topic model for sentiment analysis. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 375–384, New York, NY, USA, 2009. ACM.

[89] Andrew Lipsman. Online consumer-generated reviews have significant impact on offline purchase behavior. Technical report, Comscore Inc., 2007.

[90] Bing Liu. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*. Data-Centric Systems and Applications. Springer, 2007.

[91] Bing Liu. *Sentiment analysis and subjectivity*. Handbook of Natural Language Processing, second edition, 2010.

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

[93] Bing Liu, Minqing Hu, and Junsheng Cheng. Opinion observer: analyzing and comparing opinions on the web. In *Proceedings of the 14th international conference on World Wide Web*, WWW '05, pages 342–351, New York, NY, USA, 2005. ACM.

[94] Jingjing Liu, Yunbo Cao, Chin-Yew Lin, Yalou Huang, and Ming Zhou. Low-quality product review detection in opinion summarization. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL '07)*, pages 334–342, 2007.

[95] Elena Lloret, Alexandra Balahur, Manuel Palomar, and Andrés Montoyo. Towards a unified approach for opinion question answering and summarization. In *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, WASSA '11, pages 168–174, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[96] Y W Lo and V Potdar. A review of opinion mining and sentiment classification framework in social networks. *2009 3rd IEEE International Conference on Digital Ecosystems and Technologies*, pages 396–401, 2009.

[97] Chong Long, Jie Zhang, and Xiaoyan Zhut. A review selection approach for accurate feature rating estimation. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 766–774, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[98] Yue Lu, Malu Castellanos, Umeshwar Dayal, and ChengXiang Zhai. Automatic construction of a context-aware sentiment lexicon: an optimization approach. In *Proceedings of the 20th international conference on World wide web*, WWW '11, pages 347–356, New York, NY, USA, 2011. ACM.

[99] Yue Lu, Huizhong Duan, Hongning Wang, and ChengXiang Zhai. Exploiting structured ontology to organize scattered online opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING '07)*, pages 734–742, 2010.

[100] Yue Lu, Panayiotis Tsaparas, Alexandros Ntoulas, and Livia Polanyi. Exploiting social context for review quality prediction. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 691–700, New York, NY, USA, 2010. ACM.

[101] Yue Lu, ChengXiang Zhai, and Neel Sundaresan. Rated aspect summarization of short comments. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 131–140, New York, NY, USA, 2009. ACM.

[102] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to information retrieval*, pages I–XXI, 1–482. Cambridge University Press, 2008.

[103] Ryan T. McDonald, Kerry Hannan, Tyler Neylon, Mike Wells, and Jeffrey C. Reynar. Structured models for fine-to-coarse sentiment analysis. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL '07)*, 2007.

[104] Qiaozhu Mei, Xu Ling, Matthew Wondra, Hang Su, and ChengXiang Zhai. Topic sentiment mixture: modeling facets and opinions in weblogs. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 171–180, New York, NY, USA, 2007. ACM.

[105] Prem Melville, Wojciech Gryc, and Richard D. Lawrence. Sentiment analysis of blogs by combining lexical knowledge with text classification. In *Proceedings of the 15th ACM*

SIGKDD international conference on Knowledge discovery and data mining, KDD '09, pages 1275–1284, New York, NY, USA, 2009. ACM.

[106] Xinfan Meng and Houfeng Wang. Mining user reviews: from specification to summarization. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, ACLShort '09, pages 177–180, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[107] Qingliang Miao, Qiudan Li, and Ruwei Dai. A unified framework for opinion retrieval. In *Proceedings of Web Intelligence*, pages 739–742, 2008.

[108] Samaneh Moghaddam and Martin Ester. The FLDA model for aspect-based opinion mining: Addressing the cold start problem. In *To appear in Proceedings of the 22nd International World Wide Web Conference (WWW '13)*.

[109] Samaneh Moghaddam and Martin Ester. Opinion digger: an unsupervised opinion miner from unstructured product reviews. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1825–1828, New York, NY, USA, 2010. ACM.

[110] Samaneh Moghaddam and Martin Ester. Aqa: Aspect-based opinion question answering. In *Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops*, ICDMW '11, pages 89–96, Washington, DC, USA, 2011. IEEE Computer Society.

[111] Samaneh Moghaddam and Martin Ester. Ilda: interdependent lda model for learning latent aspects and their ratings from online product reviews. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, pages 665–674, New York, NY, USA, 2011. ACM.

[112] Samaneh Moghaddam and Martin Ester. Aspect-based opinion mining from product reviews. In *Proceedings of the 35th International ACM SIGIR conference on research and development in Information Retrieval (SIGIR '12)*, page 1184, 2012.

[113] Samaneh Moghaddam and Martin Ester. On the design of lda models for aspect-based opinion mining. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, CIKM '12, pages 803–812, New York, NY, USA, 2012. ACM.

[114] Samaneh Moghaddam and Martin Ester. Aspect-based opinion mining in online reviews: A survey. *Submitted to IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 2013.

[115] Samaneh Moghaddam, Mohsen Jamali, and Martin Ester. Review recommendation: personalized prediction of the quality of online reviews. In *Proceedings of the 20th ACM Conference on Information and Knowledge Management (CIKM '11)*, pages 2249–2252, 2011.

[116] Samaneh Moghaddam, Mohsen Jamali, and Martin Ester. Etf: extended tensor factorization model for personalizing prediction of review helpfulness. In *Proceedings of the fifth ACM international conference on Web search and data mining*, WSDM '12, pages 163–172, New York, NY, USA, 2012. ACM.

[117] Saif Mohammad, Cody Dunne, and Bonnie Dorr. Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2*, EMNLP '09, pages 599–608, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[118] Arjun Mukherjee and Bing Liu. Aspect extraction through semi-supervised modeling. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 339–348, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[119] Arjun Mukherjee, Bing Liu, Junhui Wang, Natalie Glance, and Nitin Jindal. Detecting group review spam. In *Proceedings of the 20th international conference companion on World wide web*, WWW '11, pages 93–94, New York, NY, USA, 2011. ACM.

[120] Seung-Hoon Na, Yeha Lee, Sang-Hyob Nam, and Jong-Hyeok Lee. Improving opinion retrieval based on query-specific sentiment lexicon. In *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, ECIR '09, pages 734–738, Berlin, Heidelberg, 2009. Springer-Verlag.

[121] Thao Pham Thanh Nguyen, Takahiro Hayashi, Rikio Onai, Yuhei Nishioka, Takamasa Takenaka, and Masaya Mori. A new minimally supervised learning method for semantic term classification - experimental results on classifying ratable aspects discussed in customer reviews. In *Proceedings of the IEEE International Conference on Data Mining Workshops (ICDM Workshops '09)*, pages 43–50, 2009.

[122] Hitoshi Nishikawa, Takaaki Hasegawa, Yoshihiro Matsuo, and Genichiro Kikui. Opinion summarization with integer linear programming formulation for sentence extraction and ordering. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 910–918, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[123] Hitoshi Nishikawa, Takaaki Hasegawa, Yoshihiro Matsuo, and Genichiro Kikui. Optimizing informativeness and readability for sentiment summarization. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, pages 325–330, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[124] Michael P. O'Mahony and Barry Smyth. Learning to recommend helpful hotel reviews. In *Proceedings of the third ACM conference on Recommender systems*, RecSys '09, pages 305–308, New York, NY, USA, 2009. ACM.

[125] Jahna Otterbacher. 'helpfulness' in online communities: a measure of message quality. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 955–964, New York, NY, USA, 2009. ACM.

[126] Sinno Jialin Pan, Xiaochuan Ni, Jian-Tao Sun, Qiang Yang, and Zheng Chen. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 751–760, New York, NY, USA, 2010. ACM.

[127] Bo Pang and Lillian Lee. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 115–124, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[128] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135, January 2008.

[129] Bo Pang and Lillian Lee. Using very simple statistics for review search: An exploration. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING '08)*, pages 75–78, 2008.

[130] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[131] Isidro Peñalver Martínez, Rafael Valencia-García, and Francisco García-Sánchez. Ontology-guided approach to feature-based opinion mining. In *Proceedings of the 16th international conference on Natural language processing and information systems*, NLDB'11, pages 193–200, Berlin, Heidelberg, 2011. Springer-Verlag.

[132] Ana-Maria Popescu and Oren Etzioni. Extracting product features and opinions from reviews. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 339–346, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[133] M. F. Porter. An algorithm for suffix stripping. *Readings in information retrieval*, 1997.

[134] Katharina Probst, Rayid Ghani, Marko Krema, Andrew Fano, and Yan Liu. Semi-supervised learning of attribute-value pairs from product descriptions. In *Proceedings of the 20th international joint conference on Artifical intelligence*, IJCAI'07, pages 2838–2843, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[135] Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. Expanding domain sentiment lexicon through double propagation. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI '09)*, pages 1199–1204, 2009.

[136] Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. Opinion word expansion and target extraction through double propagation. *Computational Linguistics*, 37(1):9–27, 2011.

[137] Lizhen Qu, Georgiana Ifrim, and Gerhard Weikum. The bag-of-opinions method for review rating prediction from sparse text patterns. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 913–921, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[138] Santosh Raju, Prasad Pingali, and Vasudeva Varma. An unsupervised approach to product attribute extraction. In *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, ECIR '09, pages 796–800, Berlin, Heidelberg, 2009. Springer-Verlag.

[139] William M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

[140] Robert Remus and Christian Hänig. Towards well-grounded phrase-level polarity analysis. In *Proceedings of the 12th international conference on Computational linguistics and intelligent text processing - Volume Part I*, CICLing '11, pages 380–392, Berlin, Heidelberg, 2011. Springer-Verlag.

[141] Ellen Riloff, Siddharth Patwardhan, and Janyce Wiebe. Feature subsumption for opinion analysis. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP '06)*, pages 440–448, 2006.

[142] Ellen Riloff and Janyce Wiebe. Learning extraction patterns for subjective expressions. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, EMNLP '03, pages 105–112, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[143] Ellen Riloff, Janyce Wiebe, and Theresa Wilson. Learning subjective nouns using extraction pattern bootstrapping. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 25–32, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[144] Ruslan Salakhutdinov and Andriy Mnih. Probabilistic matrix factorization. In *Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems (NIPS '07)*, 2007.

[145] Franco Salvetti, Stephen Lewis, and Christoph Reichenbach. Automatic opinion polarity classification of movie reviews. *Colorado Research in Linguistics*, 2004.

[146] Christina Sauper, Aria Haghighi, and Regina Barzilay. Content models with attitude. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 350–358, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[147] Christopher Scaffidi, Kevin Bierhoff, Eric Chang, Mikhael Felker, Herman Ng, and Chun Jin. Red opal: product-feature scoring from reviews. In *Proceedings of the 8th ACM conference on Electronic commerce*, EC '07, pages 182–191, New York, NY, USA, 2007. ACM.

[148] Gideon Schwarz. Estimating the dimension of a model. *The Annal of Statistics*, 6(2):461–464, 1978.

[149] Shabnam Shariaty and Samaneh Moghaddam. Fine-grained opinion mining using conditional random fields. In *Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops*, ICDMW '11, pages 109–114, Washington, DC, USA, 2011. IEEE Computer Society.

[150] Kazutaka Shimada and Tsutomu Endo. Seeing several stars: a rating inference task for a document containing several evaluation criteria. In *Proceedings of the 12th Pacific-Asia conference on Advances in knowledge discovery and data mining*, PAKDD'08, pages 1006–1014, Berlin, Heidelberg, 2008. Springer-Verlag.

[151] Stefan Siersdorfer, Sergiu Chelaru, Wolfgang Nejdl, and Jose San Pedro. How useful are your comments?: analyzing and predicting youtube comments and comment ratings. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 891–900, New York, NY, USA, 2010. ACM.

[152] Benjamin Snyder and Regina Barzilay. Multiple aspect ranking using the good grief algorithm. In *Proceedings of Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL*, pages 300–307, 2007.

[153] Swapna Somasundaran, Theresa Wilson, Janyce Wiebe, and Veselin Stoyanov. Qa with attitude: Exploiting opinion type analysis for improving question answering in on-line discussions and the news. In *Proceedings of the First International Conference on Weblogs and Social Media (ICWSM '07)*, 2007.

[154] Ramakrishnan Srikant and Rakesh Agrawal. Mining sequential patterns: Generalizations and performance improvements. In *Proceedings of the 5th International Conference on Extending Database Technology: Advances in Database Technology*, EDBT '96, pages 3–17, London, UK, UK, 1996. Springer-Verlag.

[155] Philip J. Stone and Earl B. Hunt. A computer approach to content analysis: studies using the general inquirer system. In *Proceedings of the joint computer conference*, AFIPS '63 (Spring), pages 241–256, New York, NY, USA, 1963. ACM.

[156] Veselin Stoyanov, Claire Cardie, and Janyce Wiebe. Multi-perspective question answering using the opqa corpus. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 923–930, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[157] Qi Su, Xinying Xu, Honglei Guo, Zhili Guo, Xian Wu, Xiaoxun Zhang, Bin Swen, and Zhong Su. Hidden sentiment association in chinese web opinion mining. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 959–968, New York, NY, USA, 2008. ACM.

[158] Maggy Anastasia Suryanto, Ee Peng Lim, Aixin Sun, and Roger H. L. Chiang. Quality-aware collaborative question answering: methods and evaluation. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pages 142–151, New York, NY, USA, 2009. ACM.

[159] Maggy Anastasia Suryanto, Ee Peng Lim, Aixin Sun, and Roger H. L. Chiang. Quality-aware collaborative question answering: methods and evaluation. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pages 142–151, New York, NY, USA, 2009. ACM.

[160] Oscar Täckström and Ryan McDonald. Discovering fine-grained sentiment with latent variable structured prediction models. In *Proceedings of the 33rd European conference on Advances in information retrieval*, ECIR '11, pages 368–374, Berlin, Heidelberg, 2011. Springer-Verlag.

[161] Oscar Täckström and Ryan McDonald. Semi-supervised latent variable models for sentence-level sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2*, HLT '11, pages 569–574, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[162] Huifeng Tang, Songbo Tan, and Xueqi Cheng. A survey on sentiment detection of reviews. *Expert Syst. Appl.*, 36(7):10760–10773, September 2009.

[163] Ivan Titov and Ryan McDonald. Modeling online reviews with multi-grain topic models. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 111–120, New York, NY, USA, 2008. ACM.

[164] Ivan Titov and Ryan T. McDonald. A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL '8)*, pages 308–316, 2008.

[165] Panayiotis Tsaparas, Alexandros Ntoulas, and Evimaria Terzi. Selecting a comprehensive set of reviews. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '11, pages 168–176, New York, NY, USA, 2011. ACM.

[166] Peter D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 417–424, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[167] Peter D. Turney and Michael L. Littman. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Trans. Inf. Syst.*, 21(4):315–346, October 2003.

[168] S. Vegnaduzzo. Acquisition of subjective adjectives with limited resources. In *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications (2004).*

[169] Dingding Wang, Shenghuo Zhu, Tao Li, and Yihong Gong. Comparative document summarization via discriminative sentence selection. volume 6, pages 12:1–12:18, New York, NY, USA, October 2012. ACM.

[170] Hongning Wang, Yue Lu, and Chengxiang Zhai. Latent aspect rating analysis on review text data : A rating regression approach. pages 783–792. ACM, 2010.

[171] Hongning Wang, Yue Lu, and ChengXiang Zhai. Latent aspect rating analysis without aspect keyword supervision. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '11)*, pages 618–626, 2011.

[172] Xuerui Wang and Andrew McCallum. Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '06, pages 424–433, New York, NY, USA, 2006. ACM.

[173] Wei Wei and Jon Atle Gulla. Sentiment learning on product reviews via sentiment ontology tree. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 404–413, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[174] Markus Weimer, Iryna Gurevych, and Max Mühlhäuser. Automatically assessing the post quality in online discussions on software. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 125–128, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.

[175] Janyce M. Wiebe, Rebecca F. Bruce, and Thomas P. O'Hara. Development and use of a gold-standard data set for subjectivity classifications. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pages 246–253, Stroudsburg, PA, USA, 1999. Association for Computational Linguistics.

[176] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 347–354, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[177] Theresa Wilson, Janyce Wiebe, and Rebecca Hwa. Just how mad are you? finding strong and weak opinion clauses. In *Proceedings of the 19th national conference on Artifical intelligence*, AAAI'04, pages 761–767. AAAI Press, 2004.

[178] Tak-Lam Wong, Lidong Bing, and Wai Lam. Normalizing web product attributes and discovering domain ontology with minimal effort. In *Proceedings of the Forth International Conference on Web Search and Web Data Mining (WSDM '11)*, pages 805–814, 2011.

[179] Tak-Lam Wong, Wai Lam, and Tik-Shun Wong. An unsupervised framework for extracting and normalizing product attributes from multiple web sites. In *Proceedings of the 31st annual*

international ACM SIGIR conference on Research and development in information retrieval, SIGIR '08, pages 35–42, New York, NY, USA, 2008. ACM.

[180] Jeonghee Yi, Tetsuya Nasukawa, Razvan Bunescu, and Wayne Niblack. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In *Proceedings of the Third IEEE International Conference on Data Mining*, ICDM '03, pages 427–, Washington, DC, USA, 2003. IEEE Computer Society.

[181] Hong Yu and Vasileios Hatzivassiloglou. Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, EMNLP '03, pages 129–136, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[182] Jianxing Yu, Zheng-Jun Zha, Meng Wang, and Tat-Seng Chua. Aspect ranking: identifying important product aspects from online consumer reviews. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1496–1505, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[183] Jianxing Yu, Zheng-Jun Zha, Meng Wang, Kai Wang, and Tat-Seng Chua. Domain-assisted product aspect hierarchy generation: towards hierarchical organization of unstructured consumer reviews. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 140–150, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[184] Lei Yu Lei Yu, Jia Ma Jia Ma, Seiji Tsuchiya Seiji Tsuchiya, and Fuji Ren Fuji Ren. Opinion mining: A study on semantic orientation analysis for online document. pages 4548–4552. IEEE, 2008.

[185] Xiaohui Yu, Yang Liu, Xiangji Huang, and Aijun An. A quality-aware model for sales prediction using reviews. In *Proceedings of the 19th International Conference on World Wide Web (WWW '10)*, pages 1217–1218, 2010.

[186] Zhongwu Zhai, Bing Liu, Hua Xu, and Peifa Jia. Grouping product features using semi-supervised learning with soft-constraints. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 1272–1280, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[187] Zhongwu Zhai, Bing Liu, Hua Xu, and Peifa Jia. Constrained lda for grouping product features in opinion mining. In *Proceedings of the 15th Pacific-Asia Conference-Advances in Knowledge Discovery and Data Mining (PAKDD '11)*, pages 448–459, 2011.

[188] Tian-Jie Zhan and Chun-Hung Li. Semantic dependent word pairs generative model for fine-grained product feature mining. In *Proceedings of the 15th Pacific-Asia conference on Advances in knowledge discovery and data mining - Volume Part I*, PAKDD'11, pages 460–475, Berlin, Heidelberg, 2011. Springer-Verlag.

[189] Lei Zhang and Bing Liu. Identifying noun product features that imply opinions. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2*, HLT '11, pages 575–580, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[190] Lei Zhang, Bing Liu, Suk Hwan Lim, and Eamonn O'Brien-Strain. Extracting and ranking product features in opinion documents. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 1462–1470, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[191] Min Zhang and Xingyao Ye. A generation model to unify topic relevance and lexicon-based sentiment for opinion retrieval. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '08)*, pages 411–418, 2008.

[192] Wei Zhang, Clement Yu, and Weiyi Meng. Opinion retrieval from blogs. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, CIKM '07, pages 831–840, New York, NY, USA, 2007. ACM.

[193] Zhu Zhang and Balaji Varadarajan. Utility scoring of product reviews. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, CIKM '06, pages 51–57, New York, NY, USA, 2006. ACM.

[194] Wayne Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. Jointly modeling aspects and opinions with a maxent-lda hybrid. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 56–65, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[195] Yanyan Zhao, Bing Qin, Shen Hu, and Ting Liu. Generalizing syntactic structures for product attribute candidate extraction. In *Proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 377–380, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[196] Jingbo Zhu, Huizhen Wang, Benjamin K. Tsou, and Muhua Zhu. Multi-aspect opinion polling from textual reviews. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 1799–1802, New York, NY, USA, 2009. ACM.

[197] Li Zhuang, Feng Jing, and Xiao-Yan Zhu. Movie review mining and summarization. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, CIKM '06, pages 43–50, New York, NY, USA, 2006. ACM.

Index