# SENTIMENT-ALIGNED TOPIC MODELS FOR PRODUCT ASPECT RATING PREDICTION

by

Hao Wang

B.Eng., Beijing University of Posts and Telecommunications, 2012

Thesis Submitted in Partial Fulfillment
of the Requirements for the Degree of

Master of Science

in the
School of Computing Science
Faculty of Applied Sciences


SIMON FRASER UNIVERSITY
Spring 2015

## APPROVAL

**Name:** Hao Wang

**Degree:** Master of Science

**Title:** SENTIMENT-ALIGNED TOPIC MODELS FOR PRODUCT ASPECT RATING PREDICTION

**Examining Committee:**

**Chair**: Dr. Ke Wang
Professor

---

**Dr. Martin Ester**
Senior Supervisor
Professor

---

**Dr. Anoop Sarkar**
Supervisor
Associate Professor

---

**Dr. Fred Popowich**
Internal Examiner
Professor

**Date Approved:** March 23rd, 2015

## Partial Copyright Licence

# Abstract

Aspect-based opinion mining has attracted lots of attention today. In this thesis, we address the problem of product aspect rating prediction, where we would like to extract the product aspects, and predict aspect ratings simultaneously. Topic models have been widely adapted to jointly model aspects and sentiments, but existing models may not do the prediction task well due to their weakness in sentiment extraction. The sentiment topics usually do not have clear correspondence to commonly used ratings, and the model may fail to extract certain kinds of sentiments due to skewed data. To tackle this problem, we propose a sentiment-aligned topic model(SATM), where we incorporate two types of external knowledge: product-level overall rating distribution and word-level sentiment lexicon. Experiments on real dataset demonstrate that SATM is effective on product aspect rating prediction, and it achieves better performance compared to the existing approaches.

**Keywords**: Opinion Mining; Topic Models; Aspect Rating Prediction; Text Mining; Sentiment Extraction

# Acknowledgements

First and foremost, I would like to thank my advisor, professor Martin Ester, who has been very supportive during my Master studies. I feel grateful for his understanding, patience and help, and I learned a lot from him in the last three years. I could not have imagined having a better advisor. I would also like to thank professor Anoop Sarkar and professor Fred Popowich for their helpful suggestions and insightful feedback on my thesis.

I have met lots of interesting people since I moved to Vancouver. They made my life more enjoyable. I really appreciate their help, especially colleagues in the Database and Data Mining lab.

Last but not least, I would like to express my gratitude to my family, who have been supportive always. I would not be able to go this far without them.

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

With the growth of social media services, a huge amount of opinionated text is generated. For example, people can write reviews to evaluate products or businesses. This opinionated text has become an important source of information. Customers tend to read online reviews to help them compare products, and make informed decisions. For businesses, they can track the feedback from customers and better market their products. However, as the volume of opinionated text continues to grow, it is often impossible to read all of them, which calls for efficient methods for opinion mining. Opinion Mining, or sentiment analysis, is the field of study that analyzes people's opinions, sentiments, evaluations, and attitudes towards entities such as products[30].

Opinion mining has been investigated mainly in three levels: document, sentence and aspect[30], although problems at different levels can be related. Common document-level tasks are document sentiment classification, review helpfulness estimation, spam review detection, etc. Tasks at the next level analyze sentences. For example, sentence subjectivity classification tries to distinguish subjective sentences from objective ones, and opinion summarization selects representative sentences as a summary.

In recent years, aspect-based opinion mining[18][37] has drawn lots of attention. An aspect refers to a rateable feature, such as *staff* and *location* for hotel, or *size* and *battery* for digital camera. Aspect-based opinion mining provides fine grained analysis, as people may express different sentiments on different aspects. For example, in Figure 1.1, the author praises the check-in service, location, but evaluates the room quality as average. These details will be lost if we only know the overall rating of the review.


Figure 1.1: An example hotel review from TripAdvisor

In this thesis, we deal with the problem of *product aspect rating prediction*. The input is a collection of products, and each product is associated with a set of reviews. The goal is to extract the aspects and predict the aspect ratings for each product. These aspect ratings can be seen as a compact summarization for the product. It will help users efficiently digest the reviews, and gain more insight into the product quality. For example, if a hotel only receives average feedback, the aspect ratings can be used as an explanation. People may praise the room quality, but complain about the staff service, so the hotel has a high aspect rating on room quality, but a low rating on the staff aspect. Such information can be used by businesses to improve their service. It will also help customers make better choices, as customers may have different preferences on different aspects. For business travellers, the business facilities such as wifi may be of high importance, but tourists may pay more attention to the location.

Currently, certain websites already request users to provide explicit aspect ratings as feedback, such as TripAdvisor[1], BeerAdvocate[2], but on most online review websites, the aspect ratings are not available, and users only provide document-level overall rating. However, recovering these aspect ratings can be achieved by mining the online reviews, which motivates our work.

## 1.1 Challenges

The product aspect rating prediction problem usually involves two subtasks: aspect extraction and sentiment identification[55]. Given some text, we would like to know what

[1] www.tripadvisor.com

[2] www.beeradvocate.com

aspects it talks about, and what kind of sentiments are expressed. For example, given a sentence "the room is filthy", we would like to know that it talks about the aspect "room". Also, "filthy" is a sentiment word, and it expresses strongly negative sentiment towards the aspect "room". Then we can aggregate information from all sentences to learn the sentiments towards all the aspects.

Topic models[6, 17] have been popular in aspect-based opinion mining[30]. It assumes each document consists of a mixture of topics and each topic is a probability distribution over words. Existing works have used topic models to extract only aspects[54, 8, 11], or jointly model aspects and sentiments [34, 28, 26, 20, 36, 23, 50, 40, 24, 39, 21]. An aspect is usually modelled as a distribution over aspect words. For example, an aspect on room quality may have high probability over words "room", "bathroom", "bed", while an aspect on staff may have high probability over words "reservation", "service". Similarly, a sentiment topic is modelled as a distribution over sentiment words. A sentiment topic having high probability mass on words "wonderful", "excellent" expresses highly positive sentiment, while a topic with high frequency words "OK", "average" is less positive. If the model is to be applied for aspect rating prediction, the sentiment topics should have clear correspondence to the ratings, so the model sentiments are accordant with the "true" human sentiments[25]. In other words, the sentiment topics, unlike aspects, need to be ordered[23].

To illustrate this, suppose there are 5 sentiment topics. We refer to its index as *sentiment label*, so the sentiment labels are from 1 to 5. The sentiment topic with label $i$ is expected to correspond to the rating $i$ on the 1-5 rating scale. For example, the sentiment topic with label 5 should have high probability over positive sentiment words, so it expresses highly positive sentiment, which matches our natural interpretation for the rating 5. In this case, sentiment labels and ratings are *aligned*.

However, in a standard topic model, the sentiment topic extraction is unsupervised, which means the learned sentiment topics may not have clear correspondence with different ratings. Also, if the positive reviews are dominant in the data, the topic model may fail to capture the negative sentiments with any sentiment topic, so no sentiment labels are matched with low ratings. What's more, the correspondence between sentiment labels and ratings can be different at each run. If the sentiment labels are not correctly aligned to the

ratings, we cannot use these sentiment labels to predict aspect ratings, thus the methods are less practical. We call this the *sentiment label alignment* problem.

To tackle this problem, models in the literature usually take two approaches. The most common one is to use some seed words for each sentiment topic. These seed words can be used to define Dirichlet priors with asymmetric concentration parameter vectors [50, 21, 25], or to initialize word assignment to sentiment topic[28], or both[26, 20]. However, these seed words are usually arbitrarily selected, and how to define asymmetric priors is not clear, especially when we would like to capture more than two (positive and negative) kinds of sentiments. Another approach is to use the review overall ratings. For example, a regression model can be built, with features related to sentiment topics, and the overall rating is the response[23, 58, 59]. However, the regression model usually has no constraints on the sentiment topic orders. Sentiment labels and ratings still don't have direct matchings.

## 1.2 Contributions

Topic modeling provides a flexible and powerful framework to model aspects and sentiments in opinionated text. However, existing works focus more on aspect extraction, while less on sentiment identification. For the aspect rating prediction task, the prediction accuracy is usually compromised due to the sentiment label alignment problem.

In this thesis, we propose sentiment-aligned topic models(SATM), which jointly models aspects and sentiments in opinionated text. To use it for aspect rating prediction, we focus on the sentiment label alignment problem. The extracted sentiment topics can capture different kinds of sentiments, ranging from highly positive to highly negative. More importantly, the sentiment labels can be aligned with ratings, so we can directly use these sentiment labels to predict aspect ratings. To achieve this goal, we incorporate two kinds of external knowledge in the model: the product overall rating distribution, and a sentiment lexicon.

**Product overall rating distribution**. To help consumers better understand the quality of a product, many websites aggregate the review overall ratings, and display the distribution. This information is available on most online review websites. Figure 1.2 is an example


Figure 1.2: The overall rating distribution for Kindle on Amazon

for the product Kindle Reader on Amazon[3]. Although the overall rating distribution cannot provide aspect-level details about the product, it shows the big picture of the product-level sentiments. Intuitively, the aspect ratings usually do not diverge too far from the overall ratings, and we take this assumption. In SATM, for each product and each aspect, we define a multinomial distribution over sentiment labels, with prior parameterized by the overall rating distribution. As the overall rating distribution affects the prior, we assume that *before* we analyze the text, we already have some knowledge about the aspect ratings, but our knowledge can be corrected by mining the review text.

**Sentiment lexicon**. The sentiment lexicon is a list of sentiment-bearing words, and each word is associated with a sentiment polarity score. It can be constructed manually[52], or automatically, using seed words to expand the list of words [56]. Table 1.1 is an example with discrete sentiment score from [52]. To use this score in the model, we treat it as an extra word feature in a semi-supervised framework. In the case of manually constructed lexicon, it may only cover a very small set of words, but it provides word-level supervision, and ideally the sentiment polarity of other words can be learned through word cooccurrences.

To demonstrate the effectiveness of our model, we conduct experiments on real-world datasets. For quantitative analysis we evaluate the aspect rating prediction accuracy, and compare against state-of-the-art methods. For qualitative analysis, we show that our model can extract coherent aspects and sentiment topics.

[3]www.amazon.com

| Word | score | Word | score |
|---|---|---|---|
| amazing | 5 | worst | -5 |
| excellent | 5 | horrible | -5 |
| superior | 4 | insulting | -4 |
| worthy | 4 | pointless | -4 |
| accessible | 3 | obnoxious | -3 |
| appealing | 3 | ridiculous | -3 |
| pretty | 2 | excessive | -2 |
| efficient | 2 | inefficient | -2 |
| fair | 1 | discouraging | -1 |
| decent | 1 | pricey | -1 |

Table 1.1: Sentiment lexicon example

## 1.3 Outline of this thesis

Our method follows the topic modeling framework for aspect-based opinion mining, so in Chapter 2 we discuss the related work on topic modeling and aspect based opinion mining. In particular, we review existing methods on product aspect rating prediction.

in Chapter 3, we introduce the Sentiment-aligned topic models(SATM). We first describe a bag-of-phrases approach, where the reviews are parsed into phrases, and the model is built upon phrases. The generative process is presented, and we provide a detailed inference method. Then, the idea of incorporating overall rating distribution and sentiment lexicon can be extended to a bag-of-words approach.

Experiment result is shown in Chapter 4. We evaluate the aspect rating prediction accuracy, and compare against existing methods. We conclude this thesis in Chapter 5 and list several future work directions.

# Chapter 2

# Related work

The SATM model can be seen as an extension to topic models, so we first introduce the background about topic modeling. In particular, our method incorporates the product overall rating distribution and sentiment lexicon into the model, so it is closely related to topic models with observed features or domain knowledge. Then, we discuss aspect-based opinion mining, and especially existing methods for product aspect rating prediction. Most of the methods use topic modeling to discover aspects. Others rely mainly on word frequency, grammatical relations or supervised methods.

## 2.1 Topic modeling

Probabilistic topic models[17, 6] are a suite of statistical methods which allow documents to be explained by a set of *topics*. Latent Dirichlet Allocation(LDA)[6] is one of the simplest topic models. In LDA, each topic is characterized by a distribution over words, and it assumes there are $K$ topics. It is a generative model which specifies a probabilistic procedure by which documents can be generated. For a corpus of $D$ documents, suppose each document $d = 1 \dots D$ has length $N_d$, LDA assumes the following generative process:

- For topic $k = 1 \dots K$,
  - Draw the topic-word multinomial distribution $\phi_k \sim Dirichlet(\beta)$
- For document $d = 1 \dots D$,
  - Draw the document-topic multinomial distribution $\theta_d \sim Dirichlet(\alpha)$

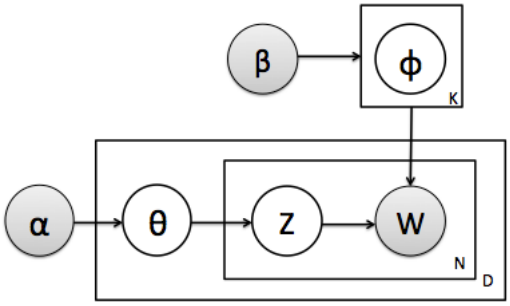


Figure 2.1: A graphical representation of LDA

- For each word $n$ in document $d$,
  1. Choose a topic $z_{d,n} \sim \theta_d$
  2. Choose a word $w_{d,n} \sim \phi_{z_{d,n}}$

LDA falls into the category of graphical models, and its graphical representation is shown in Figure 2.1. Each node is a variable and is labelled with its role in the generative process[4]. Rectangles means replication. For example, the $N$ plate indicates $N$ words in the document, and the $D$ plate indicates $D$ documents in the whole collection. The grey nodes $\alpha$, $\beta$ and $w$ are observed, while the others are latent variables.

Topic modeling is a flexible framework, and it is easy to extend the basic LDA model. Existing methods have tried to incorporate additional data to facilitate certain tasks. For example, the Author-Topic Model[47] uses author information to simultaneously model the content of documents and the interests of authors; the Topics over Time(TOT)[60] jointly models time with word cooccurrence patterns to analyze how topics evolve over time; Labeled LDA models the tags associated with documents to build the correspondence between topics and tags for *credit attribution*[46]; Andrzejewski [1] uses Dirichlet Forest Priors to express must-links and cannot-links between words. For example, we may expect that word "medication" and "medicine" should both have high or both low probability in a topic, thus constructing must-links between them. Similarly, we can define cannot-links between unrelated words, adding constraints that they cannot have both high probability in a topic.

There are also attempts to design general frameworks for incorporating observed features or domain knowledge in topic modeling. Andrzejewski[2] proposes a method to integrate any first order logic constraints, so using must-links and cannot-links is just a special case. Blei develops supervised latent Dirichlet allocation(sLDA)[5], where each document

is paired with a response. The response is general enough to be unconstrained or constrained real values, ordered or unordered class labels, and other types. Mimno points out that most topic models using metadata can fall into two categories: downstream and upstream topic models[35]. In the downstream model, it generates both the words and the metadata simultaneously given hidden topic variables. TOT and sLDA are two special cases. The upstream model can represent document-topic distributions as mixtures of metadata-specific distributions, and Author-Topic Model is an example.

## 2.2 Aspect extraction

In aspect-based opinion mining, there are four main approaches to extract explicit aspects[30]: topic modeling, word frequency based methods, grammar relation based methods and supervised learning.

### 2.2.1 Topic modeling based approaches

In topic modeling, a topic can be seen as a cluster of semantic related words. It matches the concept of *aspect* in sentiment analysis. However, when topic modeling techniques are used for aspect-based opinion mining, there are some special considerations. Titov points out that standard topic models such as PLSA and LDA usually fail to model the appropriate aspects of user reviews[54]. Instead, they propose a Multi-grain LDA (MG-LDA), using global topics and local topics to model the review content. Global topics are expected to capture general information such as properties of products, while local topics are fine-grained, meaningful aspects. However, MG-LDA uses latent aspects only to generate all the words, and does not model the sentiments in opinionated text.

The joint sentiment/topic model (JST)[28] introduces another kind of latent variable called sentiment label to model the sentiments. The word distribution is determined by both the latent aspect and sentiment label. It captures the intuition that whether a sentiment word expresses positive or negative sentiment can depend on the aspect. The *Aspect and Sentiment Unification Model*(ASUM)[20] takes a similar approach, but constrains the words in a single sentence to come from the same aspect and same sentiment label. It follows the observation that one sentence tends to represent one aspect and one sentiment, and in practice this assumption will improve the quality of extracted topics. Recent

literature tends to follow this assumption. The MaxEnt-LDA[62] further separates aspect words, sentiment words and background words. A indicator variable is used to determine the type of words. JMARS [12] takes a similar approach by using multiple types of word distributions.

Totally unsupervised methods may not extract the desired aspects, or clearly identify word types to separate aspect words and sentiment words. MaxEnt-LDA introduces a Maximum Entropy classifier, using syntax features such as POS tag to help identify the word types. The Seeded Aspect and Sentiment model (SAS)[40] is another semi-supervised approach, and it allows users to provide seed words for aspect extraction, so the aspect distributions can meet user expectations. MC-LDA[11] uses must-links and cannot-links to extract coherent aspects. Compared to the Dirichlet Forest prior approach [1], it is capable to deal with multiple word senses and the adverse effect of knowledge. The recently proposed Automated Knowledge LDA(AKL)[9] aims to learn prior knowledge automatically from a large amount of review data for aspect extraction.

There are attempts to use syntactic structures in the topic modeling framework for aspect extraction, and hidden Markov model (HMM) is widely adopted. The CFACTS-R model [23] assumes *Facet Coherence*, where users comment about a particular aspect in contiguous text fragments, and *Sentiment Coherence*, where sentiments expressed in contiguous text fragments are also related. The Dependency-Sentiment LDA[26] has a similar idea, using local conjunctive words, such as “and” or “but”, for sentiment transitions. Sauper et al.[50, 48] combine topic modeling with HMM, where the HMM part models the word type sequence. Similarly, the Joint Author Sentiment Topic Model (JAST)[41] uses HMM to capture coherence in author writing style.

Some other external information can be used for joint modeling with the review texts. In [7], reviews are associated with user-generated keyphrase lists, which are short description of pros and cons. These keyphrase lists can be used as training labels to extract semantic properties in reviews. The Twitter Opinion Topic Model (TOTM) [27] is designed for aspect extraction in tweets, and it uses hashtags, emotions as *emotion indicator*. Another commonly used external information is the overall rating associated with the text. However, it proves to be more useful for two purposes, and less about aspect extraction. First, the overall rating can be seen as user-product interaction for collaborative filtering. Existing

works have combined topic modeling(reviews) with collaborative filtering(ratings) for improved recommendation in the context of sentiment analysis[12][29]. Second, the overall ratings can help extract the sentiments in the reviews, thus being useful for aspect rating prediction. We will discuss it in the next subsection.

Other directions for aspect extraction include using multiple-word phrases instead of single words in topics[14] or capturing aspect hierarchy[21]. However, not all topic modeling approaches are built upon the original text. Instead, some of them are designed for preprocessed, intermediate representation. The interdependent LDA(ILDA)[36] assumes the text are first parsed into phrases of format $<$aspect word, sentiment word$>$. It assigns a latent aspect and latent sentiment label for each phrase. The supervised joint aspect and sentiment model (SJASM)[16], Factorized LDA[39] and TOTM[27] take the same assumption. By comparison, the Structured PLSA[32] only clusters the aspect words in the phrase. Samaneh et.al present a set of design guidelines for using topic models for aspect-based opinion mining. It shows that when separate latent variables are assumed for aspects and ratings, using such preprocessing techniques can improve the performance[38].

### 2.2.2 Other approaches

Besides topic modeling approaches, other methods for aspect extraction include frequency based methods, grammar relation based methods, supervised learning, and these methods can be combined. In this case, the extracted aspect words need to be clustered manually[61]. For example, *picture* and *photo* may refer to the same aspect in digital camera reviews. By comparison, topic modelling approaches extract aspect words and cluster them simultaneously.

Frequency based methods apply constraints on high frequency nouns to identify product aspects[18, 43, 3]. The reason is that when people talk about different aspects of a product, the vocabulary usually converges. Frequently mentioned nouns are more likely to be important aspects[30]. Various methods can be used to improve extraction accuracy. Compactness pruning and redundancy pruning are used in [18]. The OPINE system evaluates the pointwise mutual information (PMI) score between the aspect noun and some *meronymy discriminators* associated with the entity (e.g., “of scanner”, “scanner has”, “scanner comes with”, etc. for scanners)[43].

Since opinion has targets, the syntax relation between aspects and sentiment words can be used. In [64], a dependency parser was used to identify such relations for aspect extraction. This idea was further generalized into the double-propagation method for simultaneously extracting both sentiment words and aspects[44]. In [38], it demonstrates that the dependency parser technique results in the best performance to extract phrases of the format $<$aspect word, sentiment word$>$.

Aspect extraction can be seen as a special case of information extraction, and sequential learning methods can be adopted[30]. The most popular ones are based on Hidden Markov Models (HMM)[45] and Conditional Random Fields(CRF)[22]. However, these methods rely on labeled data, which may not be always available.

## 2.3 Product aspect rating prediction

Product aspect rating prediction requires not only extracting aspect, but also sentiments. Sentiment-bearing words should be identified, and their polarity should be learned.

In [32], the authors studied the problem of generating an aspect rating summary for short comments, and the model is built upon phrases. After discovering the major aspects using Structured PLSA, the phrase ratings are predicted by either Local Prediction or Global Prediction, and they are aggregated to get aspect ratings. The method in [8] also first uses topic models to find aspects. Then, for each aspect, it extracts all the relevant adjectives, and builds a conjunction graph. A label propagation algorithm[63] is used on the graph to learn the sentiment polarity score of adjective words. Although this approach is not proposed for aspect rating prediction, it can be used for this task if the polarity scores of adjective words are aggregated for each aspect. All the methods above perform aspect extraction and sentiment identification separately, while our approach takes a joint modelling approach so that different subtasks can potentially reinforce with each other. To demonstrate this, we use these methods as baselines in our experiments.

Wang et al. worked on the *Latent Aspect Rating Analysis* problem[58, 59], the task of inferring aspect ratings for each review and the relative weights reviewers have placed on each aspect. In [58], aspect keywords are provided as user input, and a two-stage method, called Latent Rating Regression(LRR), is proposed. The first stage uses a bootstrapping

algorithm to obtain more related words for each aspect, and segments the document content. In the second stage, the overall rating is "generated" as weighted combination of the latent aspect ratings, and LRR is used to infer both the weights and aspect ratings. Their follow-up work [59] does not need keyword specification from users, and replaces the bootstrapping method with a topic model. However, both methods implicitly require that each review talks about all aspects, which is not always true due to the data sparsity in online reviews.

ILDA[36] is also proposed for product aspect rating prediction. Later, it was extended to FLDA [39] to address the cold start problem, when there are few reviews associated with a product. However, neither ILDA nor FLDA deals with the sentiment label alignment problem, which limits their practicality. The method in [50, 48] follows the most common approach of using seed words to define asymmetric priors. It supports only two kinds of sentiment topics: positive and negative, while how to define asymmetric priors for more sentiment topics becomes unclear. More importantly, the prior approach may not work well in practice(see Chapter 4). There are also attempts to borrow the idea of supervised latent Dirichlet allocation(sLDA)[5], building regression models with features related to sentiment topics. Usually the overall rating can be the response[23, 58, 59]. However, the regression model usually has no constraints on the sentiment topic orders. Sentiment labels and ratings still don't have direct matchings.

Another line of research on aspect rating prediction tries to model aspect ratings in a supervised setting, where some aspect ratings are given as training instances. MAS[55] uses the text discussing an aspect to build regression models for the aspect ratings. It estimates aspect rating on the basis of overall rating and uses the words assigned to the corresponding topic to compute corrections for that aspect. [33] proposes a PALE LAGER model to incorporate given aspect ratings for aspect and sentiment topic extraction. [42] proposes a multiple-instance regression (MIR) method and assigns importance weights to sentences or paragraphs, uncovering their contribution to the aspect ratings. [51] develops Good Grief algorithm to jointly learn ranking models for different aspects by modelling the dependences among aspects via *the agreement relation*.

# Chapter 3

# Sentiment-aligned Topic Model

In this chapter, we propose Sentiment-aligned Topic Models(SATM) for product aspect rating prediction. We first formalize several key concepts, based on which we introduce *sentiment association*. Then, a bag-of-phrase model is presented. It assumes the opinion phrases are already extracted from the free text, and the model is built upon the phrases. The generative process is presented, and we give a detailed inference method. To align sentiment labels with ratings, we incorporate the product overall rating distribution and a sentiment lexicon into the model. We show that this idea can also be applied to a bag-of-words model.

## 3.1 Preliminaries

**Products**: Let $P = \{P_1, P_2, \dots\}$ be a set of products. Each product $P_i$ is associated with a set of reviews $D_i = \{d_1, d_2, \dots d_{N_i}\}$, and also an overall rating distribution $Y_i$. $Y_i$ is a multinomial distribution on $R$ ratings. Usually $R = 5$.

**Aspects**: An aspect is a rateable feature of a product, and each aspect is modelled as a distribution over aspect words. The number of aspects is predefined as $K$. We assume that products in $P$ are in the same category, so they share the same aspects.

**Sentiment topics**: A sentiment topic is modelled as a distribution over sentiment words, and each sentiment topic is associated with a sentiment label. To make it consistent with commonly used rating scales, we assume there are $R$ sentiment labels, corresponding to the $R$ ratings. The challenge is that sentiment labels with higher values are expected to be

associated with sentiment topics which express more positive sentiments, so that we can match sentiment labels with ratings.

**Phrases**: An opinion phrase $f =< h, m >$ is a pair of aspect word $h$ and sentiment word $m$, such as $< room, filthy >$[32, 36]. In the bag-of-phrase approach, for each product $P_i$, we first parse the related reviews $D_i$ into phrases $F_i$, and each product can be modelled as a bag of phrases, as in [36, 39]. In our experiment, we used dependency parser techniques for phrase extraction[38].

**Sentiment lexicon** : A sentiment lexicon $L$ is a list of sentiment words, and each word $m \in L$ is associated with a sentiment polarity score $s_m$. $s_m$ can take continuous or discrete values. Note that the lexicon $L$ may not cover all the sentiment words in the vocabulary, and in the lexicon used in Chapter 4 , it only covers a small subset of words.

## 3.2 Sentiment association

The aspect or overall rating takes $R$ values, and the polarity score in the sentiment lexicon may take a different range, so the relation between ratings and polarity scores are unknown. For example, usually $R = 5$, and the lexicon in [52] provides discrete polarity score in the range of $[-5, -1] \cup [1, 5]$. To construct the relation between rating and polarity score, a naive solution is to divide the range $[-5, -1] \cup [1, 5]$ equally, and the first part $[4, 5]$ matches rating $5$. However, this matching is arbitrary and deterministic.

If we have training instances where a sentiment word $m$ is associated with both a rating $r_m$ and polarity score $s_m$, we can build a conditional distribution T$(s_m|r_m)$, where the condition is a rating, and outcome is the polarity score. In this case, T$(s_m|r_m)$ can be interpreted as the probability of observing a polarity score $s_m$, given rating $r_m$. We refer to this probability T as *sentiment association*. This will be a key component in our model. It naturally bridges the gap between ratings and polarity scores, and captures the uncertainty in their relations. The matching is no longer deterministically and arbitrarily selected.

To get the training data, for each training instance, suppose the sentiment word is $m \in L$, we need to know its rating $r_m$ and polarity score $s_m$. $s_m$ can be retrieved directly from the sentiment lexicon, and $r_m$ can be either manually or automatically annotated. For example, suppose the word $m$ appears in review $d$, we can assign the overall rating of $d$ as its rating. In this case, each word $m \in L$ will be associated with multiple training instances that have

the same value for $s_m$ but different ratings $r_m$. We adopt this approach to automatically annotate ratings, and details are described in the Experiments section. To estimate the distribution, maximum-likelihood estimation can be used.

## 3.3 The SATM model

We introduce the bag-of-phrase Sentiment-aligned Topic Model(SATM) in this section. Its graphical representation is shown in Figure 3.1, and the mathematical notations are summarized in Table 3.1. Note that the sentiment association T is observed, because it is trained independently of the topic model part, so it can be used as a "plugin".

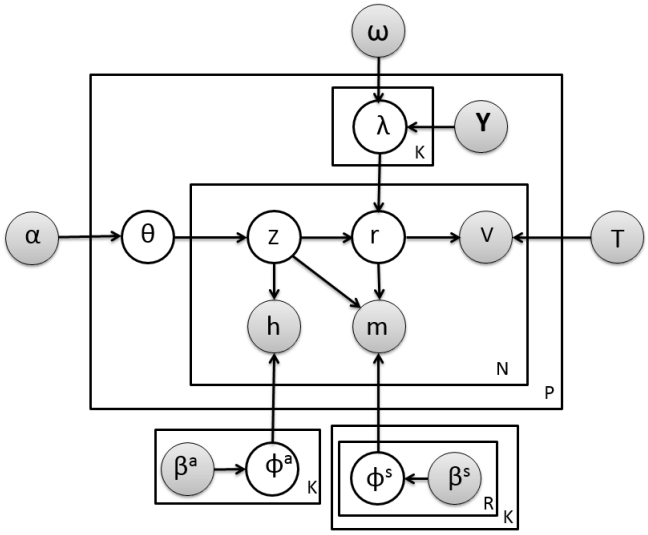


Figure 3.1: Graphical model of SATM

At the product level, each product $p$ has a multinomial distribution $\theta_p$ over aspects. It models the proportion of aspects mentioned in its reviews. For example, in the hotel domain, the reviews of hotel $A$ may talk more about the staff, when the staff aspect draws more attention, while for hotel $B$, the reviews may talk more about location. Also, for each product $p$, we define a multinomial distribution $\lambda_{p,k}$ over $R$ sentiment labels for each aspect $k$, to learn the sentiments towards different aspects. In the generative process, it captures the coherency across the reviews for a product. For example, if many reviews praise the staff of a hotel, then if another review of the same hotel talks about the staff aspect, the expressed sentiment is more likely to be positive. Also, if the sentiment labels and rating are aligned, this distribution can be used directly to predict aspect ratings.

| Symbol | Description |
|---|---|
| $\phi_k^a$ | multinomial distribution on aspect words for aspect $k$ |
| $\phi_{k,r}^s$ | multinomial distribution on sentiment words for aspect $k$ and sentiment label $r$ |
| $\beta^a, \beta^s$ | Dirichlet distribution hyperparameters for $\phi_k^a$, $\phi_{k,r}^s$ |
| $\theta_p$ | multinomial distribution on aspects for product $p$ |
| $\alpha$ | Dirichlet distribution hyperparameter for $\theta_p$ |
| $\lambda_{p,k}$ | multinomial distribution on sentiment labels for product $p$ and aspect $k$ |
| $Y_p$ | overall rating distribution of product $p$ |
| $\boldsymbol{\omega}$ | parameters in the Dirichlet distribution for $\lambda_{p,k}$ |
| $z, r$ | latent aspect $z$, latent sentiment label $r$ |
| $h, m$ | aspect word $h$, sentiment word $m$ in a phrase |
| $v$ | extra feature associated with the phrase |
| T | sentiment association |

Table 3.1: Mathematical notations for SATM

At the corpus level, each combination of aspect $k$ and sentiment label $r$ is associated with a sentiment topic $\phi_{k,r}^s$. In other words, as there are $K$ aspects and $R$ sentiment labels, there will be $|K| \times |R|$ sentiment topics to model aspect-specific sentiment words. For example, the adjective "unpredictable" in "unpredictable steering" may have negative orientation in an automobile review, but it could also have positive orientation for "unpredictable plot" in a movie review.[28]

Each observed phrase $< h, m >$ is associated with two latent variables: aspect $z$ and sentiment label $r$. Aspect $z$ models what aspect this phrase talks about, so aspect word $h$ is drawn from $\phi_k^a$. Sentiment label $r$ determines the sentiment of $m$, and $m$ is drawn from $\phi_{k,r}^s$. If $m$ is in the sentiment lexicon, we assume $r$ is also responsible for generating a word feature $v_m$, based on the sentiment association T, and the value of $v_m$ is equal to its polarity score $s_m$ in the lexicon. In this case, the observed data becomes $(< h, m >, v_m)$, and the latent sentiment label $r$ is responsible for generating both word $m$, and word feature $v_m$. For example, for the phrase <room, filthy>, we observe a word feature $v = -5$, since the sentiment polarity score for the word "filthy" is $-5$. To see how it helps align sentiment labels with ratings, note that sentiment association T is a distribution conditioned on $ratings$, but since we use $R$ sentiment labels and $R$ ratings, we build a one-to-one matching between sentiment label $i$ and rating $i$, then use $H$ conditioned on sentiment label values in the model. Suppose the observed data is $(< room, filthy >, -5)$, the word ***filthy*** is more likely to appear in reviews with low ratings such as $1$ or $2$, so given T, rating $1$ or $2$ are more likely to generate a polarity score $-5$, thus in the model sentiment labels $1$ or $2$ are more likely to

generate a word feature $-5$. The sentiment label value in the model is aligned with a rating outside of the model.

Besides, since $Y_p$ already gives us the big picture about the overall sentiment expressed on this product, we assume $\lambda_{p,k}$ is drawn from a Dirichlet distribution $Dir(\boldsymbol{\pi_{p,k}})$ with asymmetric concentration parameters, and $\pi_{p,k,r}$ is related to $Y_{p,r}$. For example, we can have $\boldsymbol{\pi_{p,k}} = f(\boldsymbol{Y_p}, \boldsymbol{\omega})$, and take a linear parametrization:

$$f(\boldsymbol{Y_p}, \boldsymbol{\omega}) = \omega_k^a \boldsymbol{Y_p} + \omega^b \tag{3.1}$$

$\omega_k^a$ captures the influence of the product overall rating distribution, and can favour certain sentiment labels in the prior. $\omega^b$ is the bias. To see how it helps align sentiment labels with ratings, for a product $p$, if its overall rating distribution $Y_p$ has high probability over rating $i$, then for any aspect $k$, we assume its product-aspect-sentiment label distribution also has high probability on sentiment label $i$ in the prior. Note that the overall rating distribution only affects the prior, and the actual aspect rating is affected by both the text which talks about aspect $k$, and also the prior. In our experiment, the parameters $\boldsymbol{w}$ are manually tuned, although automatically learning these parameters is also possible[35].

To sum up, we assume the generative process as follows:

- For each aspect $k = 1, 2, \ldots K$,
  - draw an aspect-word distribution $\phi_k^a \sim Dir(\boldsymbol{\beta}^a)$
  - For each sentiment label $r = 1, 2, \ldots R$, draw an aspect-sentiment label-word distribution $\phi_{k,r}^s \sim Dir(\boldsymbol{\beta}^s)$
- For each product $p \in P$,
  - draw a product-aspect distribution $\theta_p \sim Dir(\boldsymbol{\alpha})$
  - for each aspect $k$, draw a product-aspect-sentiment label distribution $\lambda_{p,k} \sim Dir(\boldsymbol{\pi_{p,k}})$ where $\boldsymbol{\pi_{p,k}} = f(\boldsymbol{Y_p}, \boldsymbol{\omega})$
- For each phrase $f =< h, m >$ of product $p$,
  1. Draw an aspect $z$ from $\theta_p$
  2. Draw a sentiment label $r$ from $\lambda_{p,z}$

3. Draw an aspect word $h$ from $\phi_z^a$
4. Draw a sentiment word $m$ from $\phi_{z,r}^s$. If $m \in L$, generate a word feature $v_m$ based on T.

For each product and each aspect, we can estimate a multinomial distribution over sentiment labels $\lambda_{p,k}$. Since sentiment labels and ratings are aligned, this distribution can also be seen as a distribution over ratings. As our main task is do aspect rating prediction, the aspect rating $t_{pk}$ of product $p$ on aspect $k$ can be simply calculated as the expectation of $\boldsymbol{\lambda_{p,k}}$:

$$t_{pk} = \sum_r \lambda_{p,k,r} \cdot r \tag{3.2}$$

## 3.4 Inference

We use $\Delta$ to denote the observed parameters $\boldsymbol{\alpha}, \boldsymbol{\beta^a}, \boldsymbol{\beta^s}, \boldsymbol{\pi}, \mathsf{T}$, and the posterior distribution to be estimated is $P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{\phi^s}, \boldsymbol{\phi^a} | \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta)$. Based on Bayes Theorem,

$$P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{\phi^s}, \boldsymbol{\phi^a} | \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta) = \frac{P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{\phi^s}, \boldsymbol{\phi^a}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta)}{P(\boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta)} \tag{3.3}$$

The numerator is the joint probability of latent variables and observations, and the denominator is the marginal distribution of the observations. In theory, the denominator can be calculated by summing up all the joint possibilities over all the possible latent variable combinations:

$$P(\boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta) = \sum\nolimits_{\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{\phi^s}, \boldsymbol{\phi^a}} P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{\phi^s}, \boldsymbol{\phi^a}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta) \tag{3.4}$$

However, the number of possible latent variable combinations is exponentially large, so the denominator in 3.3 is computationally hard to calculate. Instead, we use collapsed Gibbs Sampling[15] to estimate the posterior distribution $P(\boldsymbol{z}, \boldsymbol{r} | \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta)$. Note that ($\boldsymbol{\theta}$, $\boldsymbol{\lambda}$, $\boldsymbol{\phi^s}$, $\boldsymbol{\phi^a}$) are not included, because they can be integrated out for sampling. Later these distributions can be estimated given samples of $\boldsymbol{z}, \boldsymbol{r}$, as shown below.

**Joint distribution**. Given the graphical model, the joint distribution with latent variables and observed data $P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v} | \boldsymbol{\alpha}, \boldsymbol{\beta^a}, \boldsymbol{\beta^s}, \boldsymbol{\pi}, \mathsf{T})$ can be factorized as:

$$P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v} | \boldsymbol{\alpha}, \boldsymbol{\beta^a}, \boldsymbol{\beta^s}, \boldsymbol{\pi}, \mathsf{T}) = P(\boldsymbol{z} | \boldsymbol{\alpha})\, P(\boldsymbol{r} | \boldsymbol{z}, \boldsymbol{\pi})\, P(\boldsymbol{h} | \boldsymbol{z}, \boldsymbol{\beta^a})\, P(\boldsymbol{m} | \boldsymbol{z}, \boldsymbol{r}, \boldsymbol{\beta^s})\, P(\boldsymbol{v} | \boldsymbol{r}, \mathsf{T}) \tag{3.5}$$

We consider each part separately.

$$P(\boldsymbol{z}|\boldsymbol{\alpha}) = \prod_p \int_{\theta_p} P(\boldsymbol{z}|\boldsymbol{\theta_p}) P(\boldsymbol{\theta_p}|\boldsymbol{\alpha}) \, \mathrm{d}\boldsymbol{\theta_p} \tag{3.6}$$

$P(\boldsymbol{z}|\boldsymbol{\theta_p})$ is a multinomial distribution, and $P(\boldsymbol{\theta_p}|\boldsymbol{\alpha})$ is a Dirichlet distribution. Suppose $n_{p,k}$ is the number of phrases in product $p$ with latent aspect $k$, from 3.6 we can get:

$$\begin{aligned} P(\boldsymbol{z}|\boldsymbol{\alpha}) &= \prod_p \int_{\theta_p} (\prod_{k=1}^{K} \theta_{p,k}^{n_{p,k}}) (\frac{1}{B(\boldsymbol{\alpha})} \prod_{k=1}^{K} \theta_{p,k}^{\alpha_k - 1}) \, \mathrm{d}\boldsymbol{\theta_p} \\ &= \prod_p \frac{1}{B(\boldsymbol{\alpha})} \int_{\theta_p} \prod_{k=1}^{K} \theta_{p,k}^{n_{p,k} + \alpha_k - 1} \, \mathrm{d}\boldsymbol{\theta_p} \\ &= \prod_p \frac{B(\boldsymbol{n_p} + \boldsymbol{\alpha})}{B(\boldsymbol{\alpha})} \end{aligned} \tag{3.7}$$

In 3.7, $B(x)$ is the Beta function. For example, for the K-dimensional $\boldsymbol{\alpha}$

$$B(\boldsymbol{\alpha}) = \frac{\prod_{k=1}^{K} \Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^{K} \alpha_k)} \tag{3.8}$$

$\Gamma$ is the gamma function, and it satisfies $\Gamma(t+1) = t\Gamma(t)$.

Similarly, suppose $n_{k,h}^a$ is the number of times aspect word $h$ has latent aspect $k$; $n_{k,r,m}^s$ is the number of times sentiment word $m$ has latent aspect $k$ and sentiment label $r$; $n_{p,k,r}$ is the number of phrases from product $p$ with aspect $k$ and sentiment label $r$, we have:

$$\begin{aligned} P(\boldsymbol{h}|\boldsymbol{z}, \boldsymbol{\beta^a}) &= \prod_k \int_{\phi_k^a} P(\boldsymbol{h}|\boldsymbol{\phi_k^a}) P(\boldsymbol{\phi_k^a}|\boldsymbol{\beta^a}) \, \mathrm{d}\boldsymbol{\phi_k^a} \\ &= \prod_k \frac{B(\boldsymbol{n_k^a} + \boldsymbol{\beta^a})}{B(\boldsymbol{\beta^a})} \end{aligned} \tag{3.9}$$

$$\begin{aligned} P(\boldsymbol{m}|\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{\beta^s}) &= \prod_k \prod_r \int_{\phi_{k,r}^s} P(\boldsymbol{m}|\boldsymbol{\phi_{k,r}^s}) P(\boldsymbol{\phi_{k,r}^s}|\boldsymbol{\beta^s}) \, \mathrm{d}\boldsymbol{\phi_{k,r}^s} \\ &= \prod_k \prod_r \frac{B(\boldsymbol{n_{k,r}^s} + \boldsymbol{\beta^s})}{B(\boldsymbol{\beta^s})} \end{aligned} \tag{3.10}$$

$$\begin{aligned} P(\boldsymbol{r}|\boldsymbol{z}, \boldsymbol{\pi}) &= \prod_p \prod_k \int_{\lambda_{p,k}} P(\boldsymbol{r}|\boldsymbol{\lambda_{p,k}}) P(\boldsymbol{\lambda_{p,k}}|\boldsymbol{\pi}) \, \mathrm{d}\boldsymbol{\lambda_{p,k}} \\ &= \prod_p \prod_k \frac{B(\boldsymbol{n_{p,k}} + \boldsymbol{\pi})}{B(\boldsymbol{\pi})} \end{aligned} \tag{3.11}$$

$P(\boldsymbol{v}|\boldsymbol{r}, \mathsf{T})$ is the probability of observing word features $v$ given their sentiment labels $r$:

$$P(\boldsymbol{v}|\boldsymbol{r}, \mathsf{T}) = \prod_m \mathsf{T}(v_m|r_m) \tag{3.12}$$

Given the results above, the joint distribution in 3.4 can be rewritten as:

$$
\begin{aligned}
&P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v} | \boldsymbol{\alpha}, \boldsymbol{\beta^a}, \boldsymbol{\beta^s}, \boldsymbol{\pi}, \mathsf{T}) \\
&= P(\boldsymbol{z}|\boldsymbol{\alpha})\, P(\boldsymbol{r}|\boldsymbol{z}, \boldsymbol{\pi})\, P(\boldsymbol{h}|\boldsymbol{z}, \boldsymbol{\beta^a})\, P(\boldsymbol{m}|\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{\beta^s})\, P(\boldsymbol{v}|\boldsymbol{r}, \mathsf{T}) \\
&= \{\prod_p \frac{B(\boldsymbol{n_p} + \boldsymbol{\alpha})}{B(\boldsymbol{\alpha})} \prod_k \frac{B(\boldsymbol{n_{p,k}} + \boldsymbol{\pi})}{B(\boldsymbol{\pi})}\} \{\prod_k \frac{B(\boldsymbol{n_k^a} + \boldsymbol{\beta^a})}{B(\boldsymbol{\beta^a})} \prod_r \frac{B(\boldsymbol{n_{k,r}^s} + \boldsymbol{\beta^s})}{B(\boldsymbol{\beta^s})}\} \prod_m \mathsf{T}(v_m | r_m)
\end{aligned}
\tag{3.13}
$$

**Full conditional**. Given the joint probability in 3.13, for phrase $i$ of product $p$, we can get the probability of its latent aspect $z_i$ and sentiment label $r_i$, conditioned on the latent variables of other phrases. Suppose the phrase is $< h, m >$, we use $\boldsymbol{r}_{-i}$ and $\boldsymbol{z}_{-i}$ to denote the latent aspect/sentiment label of all phrases except phrase $i$. Based on Bayes theorem and 3.13:

$$
\begin{aligned}
&P(z_i = k, r_i = l | \boldsymbol{z}_{-i}, \boldsymbol{r}_{-i}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta) \\
&= \frac{P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta)}{P(\boldsymbol{z}_{-i}, \boldsymbol{r}_{-i}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta)} \\
&= \frac{P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta)}{P(\boldsymbol{z}_{-i}, \boldsymbol{r}_{-i}, \boldsymbol{h}_{-i}, \boldsymbol{m}_{-i}, \boldsymbol{v}_{-i}, \Delta)\, P(h, m, v | \boldsymbol{z}_{-i}, \boldsymbol{r}_{-i}, \boldsymbol{h}_{-i}, \boldsymbol{m}_{-i}, \boldsymbol{v}_{-i}, \Delta)} \\
&\propto \frac{P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta)}{P(\boldsymbol{z}_{-i}, \boldsymbol{r}_{-i}, \boldsymbol{h}_{-i}, \boldsymbol{m}_{-i}, \boldsymbol{v}_{-i}, \Delta)} \\
&= \frac{B(\boldsymbol{n_p} + \boldsymbol{\alpha})}{[B(\boldsymbol{n_p} + \boldsymbol{\alpha})]_{-i}} \frac{B(\boldsymbol{n_{p,k}} + \boldsymbol{\pi})}{[B(\boldsymbol{n_{p,k}} + \boldsymbol{\pi})]_{-i}} \frac{B(\boldsymbol{n_k^a} + \boldsymbol{\beta^a})}{[B(\boldsymbol{n_k^a} + \boldsymbol{\beta^a})]_{-i}} \frac{B(\boldsymbol{n_{k,r}^s} + \boldsymbol{\beta^s})}{[B(\boldsymbol{n_{k,r}^s} + \boldsymbol{\beta^s})]_{-i}} g(v_m, l)
\end{aligned}
\tag{3.14}
$$

We consider each part separately. As we calculate the probability of $z_i = k$, given 3.8:

$$
\begin{aligned}
\frac{B(\boldsymbol{n_p} + \boldsymbol{\alpha})}{[B(\boldsymbol{n_p} + \boldsymbol{\alpha})]_{-i}} &= \frac{\prod_{k=1}^K \Gamma(n_{p,k} + \alpha_k)}{\Gamma(\sum_{k=1}^K (n_{p,k} + \alpha_k))} \Big[\frac{\Gamma(\sum_{k=1}^K (n_{p,k} + \alpha_k))}{\prod_{k=1}^K \Gamma(n_{p,k} + \alpha_k)}\Big]_{-i} \\
&= \frac{\Gamma(n_{p,k} + \alpha_k)}{[\Gamma(n_{p,k} + \alpha_k)]_{-i}} \frac{[\Gamma(\sum_{k=1}^K (n_{p,k} + \alpha_k))]_{-i}}{\Gamma(\sum_{k=1}^K (n_{p,k} + \alpha_k))} \\
&= \frac{n_{p,k} + \alpha_k - 1}{\sum_{k=1}^K (n_{p,k} + \alpha_k) - 1} \\
&\propto (n_{p,k} + \alpha_k - 1)
\end{aligned}
\tag{3.15}
$$

If we consider $n_{p,k}$ as the number of phrases in product $p$ with latent aspect $k$ *before* we consider phrase $i$, we will have $\frac{B(\boldsymbol{n_p} + \boldsymbol{\alpha})}{[B(\boldsymbol{n_p} + \boldsymbol{\alpha})]_{-i}} \propto (n_{p,k} + \alpha_k)$. Similarly, as we calculate

the probability of $z_i = k, r_i = l$, we have:

$$\frac{B(\boldsymbol{n_{p,k}} + \boldsymbol{\pi})}{[B(\boldsymbol{n_{p,k}} + \boldsymbol{\pi})]_{-i}} = \frac{n_{p,k,l} + \pi_{p,k,l}}{\sum_{l'}(n_{p,k,l'} + \pi_{p,k,l'})} \tag{3.16}$$

$$\frac{B(\boldsymbol{n_{k}} + \boldsymbol{\beta^a})}{[B(\boldsymbol{n_{k}} + \boldsymbol{\beta^a})]_{-i}} = \frac{n^a_{k,h} + \beta^a}{\sum_{h'}(n^a_{k,h'} + \beta^a)} \tag{3.17}$$

$$\frac{B(\boldsymbol{n_{k,r}} + \boldsymbol{\beta^s})}{[B(\boldsymbol{n_{k,r}} + \boldsymbol{\beta^s})]_{-i}} = \frac{n^s_{k,l,m} + \beta^s}{\sum_{m'}(n^s_{k,l,m'} + \beta^s)} \tag{3.18}$$

For the term $g(v_m, l)$, it is only applicable when the sentiment word $m \in L$, and in this case $g(v_m, l) = \mathsf{T}(v_m|l)$. Combining all the results above, we have the following conditional probability. The notations for 3.19 are summarized in Table 3.2, and these counts exclude assignments for the current phrase $< h, m >$.

$$\begin{aligned} &P(z_i = k, r_i = l|\boldsymbol{z}_{-i}, \boldsymbol{r}_{-i}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta) \\ &= (n_{p,k} + \alpha)\frac{n^a_{k,h} + \beta^a}{\sum_{h'}(n^a_{k,h'} + \beta^a)}\frac{n_{p,k,l} + \pi_{p,k,l}}{\sum_{l'}(n_{p,k,l'} + \pi_{p,k,l'})}\frac{n^s_{k,l,m} + \beta^s}{\sum_{m'}(n^s_{k,l,m'} + \beta^s)}g(v_m, l) \end{aligned} \tag{3.19}$$

| Symbol | Description |
|---|---|
| $n_{p,k}$ | the number of phrases from product $p$ with aspect $k$ |
| $n^a_{k,h}$ | the number of times aspect word $h$ has latent aspect $k$ |
| $n^s_{k,l,m}$ | the number of times sentiment word $m$ has latent aspect $k$ and sentiment label $l$ |
| $n_{p,k,l}$ | the number of phrases from product $p$ with aspect $k$ and sentiment label $l$ |

Table 3.2: Mathematical notations for the conditional probability

As we can see, if the sentiment word $m \in L$, when we sample the sentiment label $r$ for this phrase, the probability of generating word feature $v_m$ from $r$ is also considered. For example, the word "excellent" has a word feature value $v_m = 5$. Based on T, the probability of generating a word feature $5$ is higher for sentiment labels with larger values. If $m \notin L$, there is no $g(v_m, l)$ term, since no word feature is associated with this phrase.

It's also possible to derive the conditional probability for latent aspect or sentiment label only. Suppose at the time of sampling latent aspect, the sentiment label $r_i = l$,

$$\begin{aligned} &P(z_i = k|\boldsymbol{z}_{-i}, \boldsymbol{r}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta) \\ &\propto (n_{p,k} + \alpha)\frac{n^a_{k,h} + \beta^a}{\sum_{h'}(n^a_{k,h'} + \beta^a)}\frac{n_{p,k,l} + \pi_{p,k,l}}{\sum_{l'}(n_{p,k,l'} + \pi_{p,k,l'})}\frac{n^s_{k,l,m} + \beta^s}{\sum_{m'}(n^s_{k,l,m'} + \beta^s)} \end{aligned} \tag{3.20}$$

Given latent aspect $z_i = k$,

$$P(r_i = l|\boldsymbol{z}, \boldsymbol{r}_{-i}, \boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta) \quad \propto \frac{n_{p,k,l} + \pi_{p,k,l}}{\sum_{l'}(n_{p,k,l'} + \pi_{p,k,l'})}\frac{n^s_{k,l,m} + \beta^s}{\sum_{m'}(n^s_{k,l,m'} + \beta^s)}g(v_m, l) \tag{3.21}$$

**Gibbs Sampler**. Based on the conditional probability in 3.19, we can design a Gibbs Sampler to estimate the posterior distribution $P(\boldsymbol{z}, \boldsymbol{r}|\boldsymbol{h}, \boldsymbol{m}, \boldsymbol{v}, \Delta)$. We jointly sample the aspect $z$ and sentiment label $r$ for the $i$th phrase $< h, m >$ of product $p$, given the latent variable assignments of other phrases. We use random initialization, as the stationary distribution does not depend on the starting point. The sampler is demonstrated as follows(without the initialization part).

As we can see, for each phrase, we explore a two-dimensional space with size $|K| \times |R|$. Another approach is to do separate sampling for each phrase, alternating between sampling latent aspect and latent sentiment label, based on Equation 3.20 and 3.21. For phrase $i$, when we sample its latent aspect, we use the latest latent sentiment label $r_i$, and similarly for sampling the sentiment label.

**Algorithm 1** Gibbs sampler for SATM with joint sampling

**for** p = 1 ...P **do**
  **for** i = 1 ...$N_p$ **do**
    phrase $i =< h, m >$
    $k \leftarrow z_i$, $l \leftarrow r_i$
    $n_{p,k} \leftarrow n_{p,k} - 1$; $n^a_{k,h} \leftarrow n^a_{k,h} - 1$
    $n^s_{k,l,m} \leftarrow n^s_{k,l,m} - 1$; $n_{p,k,l} \leftarrow n_{p,k,l} - 1$
    **for** k = 1 ...$K$ **do**
      **for** l = 1 ...$R$ **do**
        Evaluate $P[k][l]$ based on Equation 3.19
      **end for**
    **end for**
    Sample $(k', l')$ based on distribution $P$
    $z_i \leftarrow k'$, $r_i \leftarrow l'$
    $n_{p,k'} \leftarrow n_{p,k'} + 1$; $n^a_{k',h} \leftarrow n^a_{k',h} + 1$
    $n^s_{k',l',m} \leftarrow n^s_{k',l',m} + 1$; $n_{p,k',l'} \leftarrow n_{p,k',l'} + 1$
  **end for**
**end for**

**Multinomial parameters**. Based on the samples $(\boldsymbol{z}, \boldsymbol{r})$, we can estimate the multinomial distributions ($\boldsymbol{\theta}$, $\boldsymbol{\lambda}$, $\boldsymbol{\phi^s}$, $\boldsymbol{\phi^a}$). We first take $\boldsymbol{\theta}$ as an example.

$$P(\theta_p|\boldsymbol{z}, \boldsymbol{\alpha}) = \frac{P(\theta_p|\boldsymbol{\alpha})\, P(\boldsymbol{z}|\theta_p)}{P(\boldsymbol{z}|\boldsymbol{\alpha})} \tag{3.22}$$

From Equation 3.7, the denominator $P(\boldsymbol{z}|\boldsymbol{\alpha}) = \frac{B(\boldsymbol{n_p}+\boldsymbol{\alpha})}{B(\boldsymbol{\alpha})}$; $P(\theta_p|\boldsymbol{\alpha})$ is a Dirichlet distribution, while $P(\boldsymbol{z}|\theta_p)$ follows multinomial distribution:

$$\begin{aligned} P(\theta_p|\boldsymbol{z},\boldsymbol{\alpha}) &= [\frac{1}{B(\boldsymbol{\alpha})}\prod_k \theta_{p,k}^{\alpha_k-1}][\prod_k \theta_{p,k}^{n_{p,k}}]\frac{B(\boldsymbol{\alpha})}{B(\boldsymbol{n_p}+\boldsymbol{\alpha})} \\ &= \frac{1}{B(\boldsymbol{n_p}+\boldsymbol{\alpha})}\prod_k \theta_{p,k}^{n_{p,k}+\alpha_k-1} \\ &= Dir(\theta_p|\boldsymbol{n_p}+\boldsymbol{\alpha}) \end{aligned} \tag{3.23}$$

Therefore, we can get the expected $\theta_p$:

$$\theta_{p,k} = \frac{n_{p,k}+\alpha_k}{\sum_k (n_{p,k}+\alpha_k)} \tag{3.24}$$

Similarly, we get the expected value of all other distributions:

$$\phi_{k,h}^a = \frac{n_{k,h}+\beta^a}{\sum_{h'}(n_{k,h'}+\beta^a)} \tag{3.25}$$

$$\phi_{k,l,m}^a = \frac{n_{k,l,m}+\beta^s}{\sum_{m'}(n_{k,l,m'}+\beta^s)} \tag{3.26}$$

$$\lambda_{p,k,r} = \frac{n_{p,k,r}+\pi_{p,k,r}}{\sum_{r'}(n_{p,k,r'}+\pi_{p,k,r'})} \tag{3.27}$$

## 3.5 Extensions

For now, SATM is built upon phrases, so a preprocessing step is required. A phrase-based model usually has faster inference, because the number of extracted phrases are much smaller than the number of words in the original text. Also, a phrase-based assumption can simplify the model, as in each phrase, both the aspect word and sentiment word are explicit. However, there are cases that the aspect word or sentiment word is not explicit. For example, in a hotel review, instead of writing "it has a good location", a user may mention "walking distance to various tourist attractions", and it's less obvious to choose an explicit aspect word or sentiment word. What's more, the phrase extraction method usually relies on syntax relations to extract the aspect word, sentiment word and the connection between them. Many online reviews have informal writing styles, so the syntax relation may not be correctly learned. In this case, the preprocessing step may fail to extract the phrase and miss the information. For example, a phrase extraction method based on dependency parser will fail to extract the phrase for the short sentence "Wifi included".

In this subsection we introduce an extension to the SATM model, which is based on words, not phrases. We assume each sentence talks about one aspect and one kind of sentiment, as in [20]. By using such constraint, improved topic extraction quality has been observed. Then, words in a sentence are categorized into three types: aspect word, sentiment word and background word. For example, in the sentence "The food was tasty", *food* can be seen as an aspect word, and *tasty* is an sentiment word. "the" and "was" do not bear useful meanings for opinion mining so they are treated as background words. Note that a sentence may have multiple aspect words, and it is possible that none of these words will be a valid aspect word in the context of phrases, but they all help identify the *latent* aspect of the sentence thus should not be ignored. For example, in the sentence above "walk distance to various tourist attractions", "distance" and "tourist attractions" can remind us that it talks about the *location* aspect, but none of these words will easily fits in a phrase. Similarly, the same observation can be applied to sentiment words.

The idea of using external knowledge to align sentiment labels with ratings can also be adopted. For each product $p$, we still define a multinomial distribution $\lambda_{p,k}$ over $R$ sentiment labels for each aspect $k$. Same as before, its prior can be parameterized by the overall rating distribution. The way to use the sentiment lexicon can be slightly different from the bag-of-phrase model. Since each sentence has one latent sentiment label, we assume the polarity score features are not generated from the phrase-level latent sentiment label, but the sentence-level one. As each sentence may have multiple words in the sentiment lexicon, the sentiment label in a sentence can generate a list of features.

To sum up, we give a generative process as below, and its graphical representation is shown in Figure 3.2.

- For each aspect $k = 1, 2, \ldots K$,
  - draw an aspect-word distribution $\phi_k^a \sim Dir(\boldsymbol{\beta}^a)$
  - For each sentiment label $r = 1, 2, \ldots R$, draw an aspect-sentiment label-word distribution $\phi_{k,r}^s \sim Dir(\boldsymbol{\beta}^s)$
- Draw a background word distribution $\phi^b \sim Dir(\boldsymbol{\beta}^b)$
- For each product $p \in P$,
  - draw a product-aspect distribution $\theta_p \sim Dir(\boldsymbol{\alpha})$

- for each aspect $k$, draw a product-aspect-sentiment label distribution $\lambda_{p,k} \sim Dir(\boldsymbol{\pi_{p,k}})$ where $\boldsymbol{\pi_{p,k}} = f(\boldsymbol{Y_p}, \boldsymbol{\omega})$

- For each sentence $s$ of product $p$,

  1. Draw an aspect $z$ from $\theta_p$
  2. Draw a sentiment label $r$ from $\lambda_{p,z}$
  3. Draw a word type distribution $\psi_s \sim Dir(\chi)$
  4. For each word $w$, draw a word type indicator $u$ from $\psi_s$
     - if $u = 0$, draw $w$ from $\phi_z^a$
     - if $u = 1$, draw $w$ from $\phi_{z,r}^s$
     - if $u = 2$, draw $w$ from $\phi^b$
  5. if $w \in L$, generate a word feature $v_m$ from T.

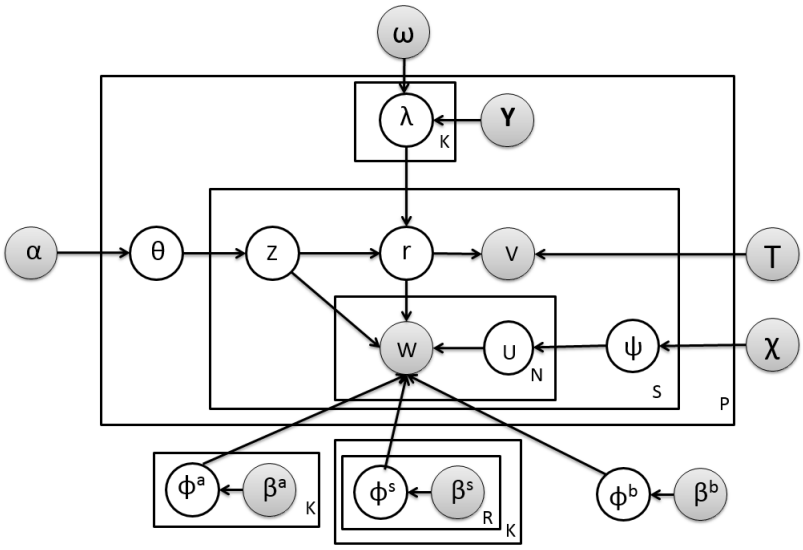


Figure 3.2: Graphical representation for the bag-of-words model

**Inference**. We can still use collapsed Gibbs Sampling for inference, as the exact posterior distribution is intractable. The inference method is similar to the one proposed in Section 3.4, but slightly different. We only highlight the difference here. The joint probability is:

$$\begin{aligned} & P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w}, \boldsymbol{u}, \boldsymbol{v} | \boldsymbol{\alpha}, \boldsymbol{\beta^a}, \boldsymbol{\beta^s}, \boldsymbol{\beta^b}, \boldsymbol{\pi}, \boldsymbol{\chi}, \mathrm{T}) \\ & = P(\boldsymbol{z}|\boldsymbol{\alpha})\, P(\boldsymbol{r}|\boldsymbol{z}, \boldsymbol{\pi})\, P(\boldsymbol{u}|\chi)\, P(\boldsymbol{w}|\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{u}, \boldsymbol{\beta^a}\boldsymbol{\beta^s}, \boldsymbol{\beta^b})\, P(\boldsymbol{v}|\boldsymbol{r}, \mathrm{T}) \end{aligned} \tag{3.28}$$

$P(\boldsymbol{z}|\boldsymbol{\alpha})$, $P(\boldsymbol{r}|\boldsymbol{z},\boldsymbol{\pi})$, $P(\boldsymbol{v}|\boldsymbol{r},\mathsf{T})$ are the same as before, except they are associated with sentences, not phrases, and $\boldsymbol{v}$ is a list of features. $P(\boldsymbol{w}|\boldsymbol{z},\boldsymbol{r},\boldsymbol{u},\boldsymbol{\beta^a}\boldsymbol{\beta^s},\boldsymbol{\beta^b})$ can be decomposed into three parts:

$$P(\boldsymbol{w}|\boldsymbol{z},\boldsymbol{r},\boldsymbol{u},\boldsymbol{\beta^a}\boldsymbol{\beta^s},\boldsymbol{\beta^b}) = P(\boldsymbol{w^a}|\boldsymbol{z},\boldsymbol{u}=\boldsymbol{0},\boldsymbol{\beta^a})\,P(\boldsymbol{w^s}|\boldsymbol{z},\boldsymbol{r},\boldsymbol{u}=\boldsymbol{1},\boldsymbol{\beta^s})\,P(\boldsymbol{w^b}|\boldsymbol{u}=\boldsymbol{2},\boldsymbol{\beta^b}) \tag{3.29}$$

In Equation 3.29, $\boldsymbol{w^a}$ is the collection of aspect words, and their word types $\boldsymbol{u}$ all equal to 0. Similarly, $\boldsymbol{w^s}$ is the collection of sentiment words, and $\boldsymbol{w^b}$ is the collection of background words.

Based on the joint probability, we still use $\Delta$ to indicate all the hyperparameters, so for each sentence $i$, we can get the full conditional probability for $(z_i = k, r_i = l)$:

$$\begin{aligned} &P(z_i = k, r_i = l|\boldsymbol{z}_{-i},\boldsymbol{r}_{-i},\boldsymbol{w},\boldsymbol{u},\boldsymbol{v},\Delta) \\ &= \frac{B(\boldsymbol{n_p}+\boldsymbol{\alpha})}{[B(\boldsymbol{n_p}+\boldsymbol{\alpha})]_{-i}}\frac{B(\boldsymbol{n_{p,k}}+\boldsymbol{\pi})}{[B(\boldsymbol{n_{p,k}}+\boldsymbol{\pi})]_{-i}}\frac{B(\boldsymbol{n_k^a}+\boldsymbol{\beta^a})}{[B(\boldsymbol{n_k^a}+\boldsymbol{\beta^a})]_{-i}}\frac{B(\boldsymbol{n_{k,r}^s}+\boldsymbol{\beta^s})}{[B(\boldsymbol{n_{k,r}^s}+\boldsymbol{\beta^s})]_{-i}}g(\boldsymbol{v_s},l) \end{aligned} \tag{3.30}$$

Although it looks exact like Equation 3.14, in the third term and fourth term, $[B(\cdot)]_{-i}$ is about excluding the $ith$ sentence, not phrase, so it means excluding multiple aspect words or sentiment words. We take the third term as an example. As we calculate the probability of $z_i = k$:

$$\begin{aligned} \frac{B(\boldsymbol{n_k^a}+\boldsymbol{\beta^a})}{[B(\boldsymbol{n_k^a}+\boldsymbol{\beta^a})]_{-i}} &= \frac{\left[\Gamma(\sum_{v=1}^{V}(n_{k,v}^a+\beta_v^a))\right]_{-i}}{\Gamma(\sum_{v=1}^{V}(n_{k,v}^a+\beta_v^a))}\frac{\prod_{v=1}^{V}\Gamma(n_{k,v}^a+\beta_v^a)}{\left[\prod_{v=1}^{V}\Gamma(n_{k,v}^a+\beta_v^a)\right]_{-i}} \\ &= \frac{\Gamma(\sum_{v=1}^{V}(n_{k,v}^a+\beta_v^a))}{\Gamma(\sum_{v=1}^{V}(n_{k,v}^a+\beta_v^a+m_{vs}^a))}\prod_{v=1}^{V}\frac{\Gamma(n_{k,v}^a+\beta_v^a+m_{vs}^a)}{\Gamma(n_{k,v}^a+\beta_v^a)} \end{aligned} \tag{3.31}$$

In Equation 3.31, $m_{vs}^a$ is the number of times word $v$ appears in sentence $s$ as an aspect word. Since a sentence only covers a few words compared to the whole vocabulary, given a sentence $s$, $m_{vs}^a$ will be $0$ for most words in the vocabulary, so the second term in 3.31 can be simplified by only considering $v$ which appears in $s$. In a similar way, we can calculate the fourth term in Equation 3.32. Suppose we use $m_{vs}^s$ to denote the number of times word $v$ appears in sentence $s$ as an sentiment word, then we have the following conditional probability:

$$
\begin{aligned}
&P(z_i = k, r_i = l | \boldsymbol{z}_{-i}, \boldsymbol{r}_{-i}, \boldsymbol{w}, \boldsymbol{u}, \boldsymbol{v}, \Delta) \\
&= (n_{p,k} + \alpha) \frac{n_{p,k,l} + \pi_{p,k,l}}{\sum_{l'}(n_{p,k,l'} + \pi_{p,k,l'})} \, g(\boldsymbol{v_s}, l) \\
&[\frac{\Gamma(\sum_{v=1}^{V}(n^a_{k,v} + \beta^a_v))}{\Gamma(\sum_{v=1}^{V}(n^a_{k,v} + \beta^a_v + m^a_{vs}))} \prod_{v=1}^{V} \frac{\Gamma(n^a_{k,v} + \beta^a_v + m^a_{vs})}{\Gamma(n^a_{k,v} + \beta^a_v)}] \\
&[\frac{\Gamma(\sum_{v=1}^{V}(n^s_{k,l,v} + \beta^s_v))}{\Gamma(\sum_{v=1}^{V}(n^s_{k,l,v} + \beta^s_v + m^s_{vs}))} \prod_{v=1}^{V} \frac{\Gamma(n^s_{k,l,v} + \beta^s_v + m^s_{vs})}{\Gamma(n^s_{k,l,v} + \beta^s_v)}]
\end{aligned} \tag{3.32}
$$

Another thing to note in 3.32 is that there is no term about background words. That's because when we sample the latent aspect and sentiment label, the word types $\boldsymbol{u}$ in a sentence are given. Since the background words are generated from $\phi^b$, a corpus level distribution, it is not related to any latent aspects or sentiment labels, so the background words will not affect this probability.

Besides the latent aspect and sentiment labels, we also need to learn the word types, which is different from the bag-of-phrase model. For a word $w_j$, suppose it is in sentence $i$, and $z_i = k, r_i = l$:

$$
\begin{aligned}
P(u_j = 0 | \boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w}, \boldsymbol{u_{-j}}, \Delta) &= \frac{\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w}, \boldsymbol{u}, \Delta)}{P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w}, \boldsymbol{u_{-j}}, \Delta)} \\
&= \frac{P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w}, \boldsymbol{u}, \Delta)}{P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w_{-j}}, \boldsymbol{u_{-j}}, \Delta) \, P(w_j | \boldsymbol{z}, \boldsymbol{r}, \boldsymbol{u_{-j}}, \Delta)} \\
&\propto \frac{P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w}, \boldsymbol{u}, \Delta)}{P(\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w_{-j}}, \boldsymbol{u_{-j}}, \Delta)} \\
&= \frac{P(\boldsymbol{u}|\chi)}{[P(\boldsymbol{u}|\chi)]_{-j}} \frac{P(\boldsymbol{w_j} | \boldsymbol{z}, \boldsymbol{r}, \boldsymbol{u}, \boldsymbol{\beta^a})}{[P(\boldsymbol{w_j} | \boldsymbol{z}, \boldsymbol{r}, \boldsymbol{u}, \boldsymbol{\beta^a})]_{-j}} \\
&= \frac{n_{s,0} + \chi_0}{\sum_{i=0}^{2}(n_{s,i} + \chi_i)} \frac{n^a_{k,w} + \beta^a_w}{\sum_{v=1}^{V}(n^a_{k,v} + \beta^a_v)}
\end{aligned} \tag{3.33}
$$

In Equation 3.33,$n_{s,0}$ is the number of aspect words in sentence $s$. Similarly, $n_{s,1}$, $n_{s,2}$ are the number of sentiment words and background words in sentence $s$, respectively.

$$
P(u_j = 1 | \boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w}, \boldsymbol{u_{-j}}, \Delta) = \frac{n_{s,1} + \chi_1}{\sum_{i=0}^{2}(n_i + \chi_i)} \frac{n^s_{k,r,w} + \beta^s_w}{\sum_{v=1}^{V}(n^s_{k,r,v} + \beta^s_v)} \tag{3.34}
$$

$$
P(u_j = 2 | \boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w}, \boldsymbol{u_{-j}}, \Delta) = \frac{n_{s,2} + \chi_2}{\sum_{i=0}^{2}(n_i + \chi_i)} \frac{n^b_w + \beta^b_w}{\sum_{v=1}^{V}(n^b_v + \beta^b_v)} \tag{3.35}
$$

Given the conditional probability 3.31, 3.33, 3.34, 3.35, a Gibbs Sampler can be built. We can iterate between sampling sentence-level latent variables and word-level word types, a method similar to [40].

**Max-Ent component**. The bag-of-word model uses a switch variable to determine word types, while totally unsupervised learning for word types based on 3.33, 3.34, 3.35 may not work well in practice[62], since topic modeling relies on word cooccurrence to extract semantic relationship between words, but word types are more about syntax roles in a sentence. To help better distinguish different word types, we can use a Max-ent classifier[62, 40]. The motivation is that aspect words, sentiment words and background words tend to play different syntax roles, e.g., they have different POS tags. For example, aspect words are more likely to be noun, while sentiment words tend to be adjectives. For word $w$, suppose the features $x_w = (POS_{w-1}, POS_w, POS_{w+1})$, the POS tags of word $w$, the word before $w$, and the word after $w$, then the probability for a certain word type is

$$P(u_w = l|x_w) = \frac{exp(q_l \cdot x_w)}{\sum_{l'} exp(q_{l'} \cdot x_w)} \tag{3.36}$$

$l$ takes value $0, 1, 2$. $q_l$ denote the MaxEnt model weights, and it can be learned from a set of training sentences with labeled background, aspect and sentiment words. One example of the labeled training instance can be ({DT,NN,VBZ}, aspect word) from the sentence "The room is huge", as the POS tags for the four words are DT, NN, VBZ, JJ respectively, according to the Stanford parser[1].

The Max-ent component can work as a "plugin", and when sampling word types, we can use:

$$P(u_j = 0|\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w}, \boldsymbol{u_{-j}}, \Delta) = \frac{exp(q_0 \cdot x_w)}{\sum_{l'} exp(q_{l'} \cdot x_w)} \frac{n^a_{k,w} + \beta^a_w}{\sum_{v=1}^{V}(n^a_{k,v} + \beta^a_v)} \tag{3.37}$$

$$P(u_j = 1|\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w}, \boldsymbol{u_{-j}}, \Delta) = \frac{exp(q_1 \cdot x_w)}{\sum_{l'} exp(q_{l'} \cdot x_w)} \frac{n^s_{k,r,w} + \beta^s_w}{\sum_{v=1}^{V}(n^s_{k,r,v} + \beta^s_v)} \tag{3.38}$$

$$P(u_j = 2|\boldsymbol{z}, \boldsymbol{r}, \boldsymbol{w}, \boldsymbol{u_{-j}}, \Delta) = \frac{exp(q_2 \cdot x_w)}{\sum_{l'} exp(q_{l'} \cdot x_w)} \frac{n^b_w + \beta^b_w}{\sum_{v=1}^{V}(n^b_v + \beta^b_v)} \tag{3.39}$$

Given the latent variable samples, we can use the same method as 3.4 to estimate the multinomial parameters.

[1] http://nlp.stanford.edu/software/lex-parser.shtml

# Chapter 4

# Experiments

In this section, we describe the experiments and analyze the results. As our task is to predict aspect ratings, in our quantitative analysis, we evaluate the aspect rating prediction accuracy, and compare against existing methods. In our qualitative analysis, we will show examples of the extracted aspects and sentiment topics.

## 4.1 Dataset

We use the TripAdvisor dataset[1][58] for evaluation. In this dataset, reviews are not only associated with overall ratings, but also with ground truth aspect ratings on 7 aspects: *value, room, location, cleanliness, check in/front desk, service, business service*. All the ratings in the dataset are in the range from 1 star to 5 stars. We first remove reviews with any missing aspect ratings as we cannot evaluate the performance without ground truth. Very short reviews(less than three sentences) are also removed, as they are usually of low quality and provide little information. The oroginal dataset contains 108891 reviews[59], and after this step the statistics of the dataset is shown in Table 4.1. Then we adopt the dependency parser technique to identify opinion phrases, and collect phrases with adjective sentiment words. The dependency parser can deal with conjunctions, negations and bigram aspect words, and it results in the best performance according to [38]. Some sample phrases are shown in Table 4.2. All words are converted into lower case, and we remove phrases containing words that appear no more than 10 times or stop words. Since we are only interested in product-level aspect rating prediction, for each product, we aggregate all

[1] http://sifaka.cs.uiuc.edu/~wang296/Data/index.html

the review overall ratings to get the overall rating distribution. The statistics of the dataset is shown in Table 4.1. The average rating is the rating averaged over all reviews and all products. As we can see, positive reviews are dominant in the data, which raises the challenge of discovering negative sentiment topics.

| #Products | #Reviews | Avg rating | #Phrases |
|---|---|---|---|
| 1850 | 61306 | 4.03 | 740982 |

Table 4.1: Statistics of the dataset

| Sentences | Phrases |
|---|---|
| The room, facing the courtyard, was large and comfortable. | <room, large>, <room, comfortable> |
| The room was not really clean. | <room, no_clean> |
| Internet access was available. | <Internet access, available> |

Table 4.2: Sample extracted phrases

## 4.2 Quantitative analysis

For quantitative analysis, we use SATM for aspect rating prediction, and compare against existing methods.

### 4.2.1 Evaluation metrics

We use three evaluation metrics for comparison.

**RMSE**: Root-mean-square error is used to measure the difference between the predicted aspect ratings and ground truth aspect ratings. It is defined as:

$$RMSE = \sqrt{\frac{\sum_p \sum_k (t_{pk} - \hat{t}_{pk})^2}{|P| \times K}} \tag{4.1}$$

where $t_{pk}$ is the predicted aspect rating for product $p$ on aspect $k$, and $\hat{t}_{pk}$ is the ground truth. RMSE is the direct evaluation on the aspect rating prediction accuracy.

**Precision@N**: For each aspect $k$, we rank the hotels based on their predicted aspect ratings, and get the top $N$ results. A hotel is considered *relevant* if its ground truth aspect rating is in the top $10\%$ of the ground truth aspect ratings of all hotels[58]. Precision@N is

defined as the percentage of the top N results that are relevant:

$$Precision@N = \frac{|\{\text{relevant hotels}\} \cap \{\text{top N ranked hotels}\}|}{N} \tag{4.2}$$

We use $N = 10$, and the result is averaged over $K$ aspects. Precision@N is used to evaluate whether the predicted aspect rating can highlight top results. This is particularly useful for search engines, where the top ones are displayed in the first page, and draw the most attention.

$\boldsymbol{\rho_{hotel}}$: Pearson correlation across hotels[58] is defined as:

$$\boldsymbol{\rho_{hotel}} = \frac{\sum_k \rho(\boldsymbol{t_k}, \hat{\boldsymbol{t}}_{\boldsymbol{k}})}{K} \tag{4.3}$$

where $\boldsymbol{t_k}$ is the predicted aspect rating vector for all hotels on aspect $k$, and $\hat{\boldsymbol{t}}_{\boldsymbol{k}}$ is the corresponding ground truth vector. $\rho(\boldsymbol{t_k}, \hat{\boldsymbol{t}}_{\boldsymbol{k}})$ is the Pearson correlation between these two vectors. It measures how the predicted ratings of aspect $k$ can preserve the order in the ground truth[58]. If we can predict an aspect-specific ranking similar to the ground truth, we can use the predicted aspect ratings to answer questions like "Is hotel $a$ better than hotel $b$ on aspect $k$?"

### 4.2.2 Baselines

We use SATM to denote the bag-of-phrase approach, and SATM-W as the bag-of-words approach. We implemented 7 baselines for comparison.

The first three baselines are **Local Prediction**, **Global Prediction** and **Graph Propagation**. They all separate aspect extraction and sentiment identification. For each phrase $f =< h, m >$ from review $d$ of product $p$, we first find the aspect assignment of this phrase. Then, we use three methods to get the phrase rating. Local Prediction[32] simply uses the overall rating of $d$ as its phrase rating. Global Prediction[32] trains a multi-class classifier to classify the sentiment word $m$ into a rating category $r \in 1, 2 \ldots R$, then assigns $r$ as the phrase rating. Graph Propagation[8] builds a conjunction graph for sentiment words, and uses a Label Propagation algorithm on the graph to learn the sentiment polarity score for each sentiment word. The score of $m$ is set as phrase rating. Finally, we aggregate all the phrases of each aspect to predict the aspect ratings. To apply these methods in our experiments, in the aspect extraction step, we adapt our model to extract only aspects, as

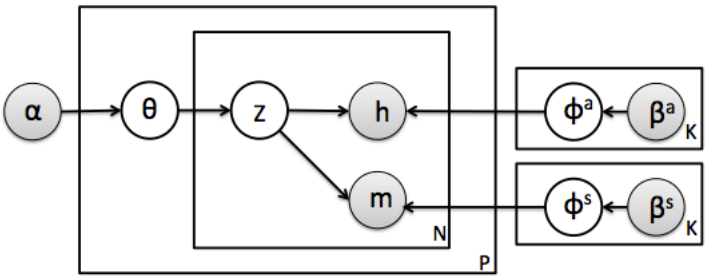


Figure 4.1: Method for aspect extraction in Local Prediction, Global Prediction and Graph Propagation

shown in Figure 4.1. In this simplified model, no sentiment labels is involved, and the latent aspect explains both the aspect word and sentiment word.

**ILDA**[36] was proposed for aspect rating prediction, but it fails to deal with the sentiment label alignment problem, so it cannot be directly used for this task. We adopt the common approach of providing seed words to set priors for each sentiment topic.

**LRR**[58] was proposed to predict aspect ratings for each review, but it can also be used to predict product aspect ratings by aggregating all the reviews of a product into a single "h-review"[59]. First, we can run a topic model to learn aspects, and annotate each sentence with an aspect. Then LRR is applied on the annotated sentences to predict aspect ratings. This approach provided the best result, according to [59]. In the first step we use the sentence-LDA[20] to annotate sentences, which is slightly different from the original method, but still provides a good analogy.

We also test two simplified versions of the SATM model. First, we remove the part which involves sentiment lexicons, so we only use the product overall rating distribution. We call this method **SATM-O**. In other words, in the generative process, we don't need to generate the word features, losing the word-level supervision. Second, we use only sentiment lexicons, ignoring the influence of overall rating distribution. We call it **SATM-L**. In this case, for each product and each aspect, the multinomial distribution over sentiment labels are drawn from a Dirichlet distribution with uniform hyperparameters. In the extreme case, in SATM, for a product, if no data is available for a particular aspect, we use the overall rating as its aspect ratings, while in SATM-L, the aspect rating distribution will be estimated to be a uniform distribution over different ratings. These two baselines can help

us identify how the sentiment lexicon and overall rating distribution can improve the results, if used separately.

Our last baseline simply uses the overall rating of a hotel as its aspect ratings. For each hotel, its overall rating is defined as the average overall rating of its reviews. This method is referred to as **Overall**. Although this method does not provide aspect-based prediction and it is usually ignored in existing literature, intuitively, in most cases the overall rating does not diverge too far from the aspect ratings.

### 4.2.3 Experimental Setup

The dataset contains ground truth aspect ratings on 7 aspects *value, room, location, cleanliness, check in/front desk, service, business service*. Therefore, for all topic modelling based approaches, the number of aspects is set to 7. Since we can evaluate aspect rating prediction only on the predefined aspects, we need to ensure the discovered aspects match the predefined aspects. To do this, we adopt the common approach of providing a few seed words for each aspect as priors, as in [58]. Since each aspect is a multinomial distribution over words, and it has Dirichlet prior parameterized by $\vec{\beta}$. We set high $\beta = 250$[40] for these seed words in the corresponding prior. For non-seed words, $\beta = 0.1$ is used. The seed words are listed in Table 4.3. There may be better methods to use seed words for aspect discovery [19, 40], and it would be interesting to combine their methods with ours. However, this is beyond the scope of this thesis, and we list it as future work. The Dirichlet prior parameter $\alpha$ for product-aspect distribution is set to $2$ in our experiment.

| **Aspects** | **Seed words** |
|---|---|
| *Value* | value, price, worth |
| *Room* | room, rooms |
| *Location* | location |
| *Cleanliness* | room, dirty, smelled, clean |
| *Check in/front desk* | staff |
| *Service* | service, breakfast, food |
| *Business service* | internet, wifi |

Table 4.3: Seed words for aspect discovery

For SATM, SATM-W, SATM-L and SATM-O, we use 5 sentiment labels, as this is the number of distinct ratings. The lexicon $L$ used in our experiment is part of [52] where words are associated with polarity scores in the range $[-5, -1] \cup [1, 5]$. We observe that words with

polarity score $1$ and $-1$ express too weak sentiments, so we discard them in our experiment. To get training instances for sentiment association T, we treat each appearance of word $m \in L$ in the data as one training instance. The polarity score $s_m$ is directly retrieved from $L$, and the rating $r_m$ is the overall rating of review $d$ where $m$ appears. This approach avoids the need for manual annotation, and the annotation result captures the characteristics of the dataset. However, all training instances in a review will have the same rating, which means that we assume all sentiment words in a review express the same sentiment, no matter what aspects they talk about. This is not true, thus will introduce noise to the training. To reduce noise, for words with positive polarity score, we ignore their appearance in reviews with rating 1 and 2, since we assume positive sentiment words rarely express negative sentiments, even if they appear in negative reviews. Therefore, T$(s_m|r_m) = 0$ for $r_m = 1, 2$ and $s_m$ in the range $[2, 5]$. Similarly, for words with negative polarity score, we ignore their appearance in reviews with rating 4 and 5. Therefore, T$(s_m|r_m) = 0$ for $r_m = 4, 5$ and $s_m$ in the range $[-5, -2]$.

For the bag-of-word approach, we prepare training instances for the Max-ent classifier. The stanford POS tagger[2] is used, and we labeled 760 words in random hotel reviews from TripAdvisor as aspect words, sentiment words or background words.

For Global Prediction, in [32], the prior for the multi-class classifier is uniform, while in our experiment, for product $p$, we used product overall rating distribution on $r$ as the prior for rating category $r$, which achieves better results than the uniform prior.

The Graph Propagation method requires a small set of sentiment words as seeds, from which the algorithm can learn sentiment score for other words. The method in [8] constructs these seed words based on morphology in an unsupervised way, and can only support two kinds of sentiment: positive and negative. In our experiment, since the sentiment lexicon is available, the sentiment seed words are from the lexicon, and we update the polarity score for those not in the lexicon.

For ILDA, since we need to provide seed words as priors for sentiment topics, we have two options, and we use both for experiment. First, we can employ the common approach of using two sentiment labels(R=2, positive and negative). Then, words with positive polarity scores in lexicon $L$ are used as priors for the positive sentiment topic, and similarly words with negative polarity scores for negative sentiment topic. An alternative approach is to

[2] http://nlp.stanford.edu/software/tagger.shtml

use 5 sentiment labels(R=5). It provides finer grained sentiment extraction, but raises the question of how to choose seed words for each sentiment topic. To do this, we use the full sentiment lexicon in [52], where sentiment words have polarity score in the range of $[-5,-1] \cup [1,5]$. We divide the lexicon, and use words with polarity score 4 and 5 as prior for the sentiment topic with label $5$. Then, words with polarity score $2$ and $3$ are used for the sentiment topic with label $4$, and so on.

For all topic modelling based approaches, we set the number of iterations for Gibbs Sampling to 3000, and take samples from the markov chain every 50 iterations after a burn-in period of 1000 iterations, as we observe the likelihood of the dataset gets stable after the 1000 iterations. In SATM, SATM-W and SATM-O, for all aspects $k$, we need to choose the parameters $\omega$. In our experiment, the parameters are manually tuned to $\omega_k^a = 20$, $w_k^b = 0.01$. These parameters can be chosen based on cross validation, and automatically learning these parameters is feasible. One possible option is to use stochastic EM sampling scheme, as in [35]. Note that we don't need to access the ground truth aspect ratings during training. For the LRR implementation[3], we use the default parameters included in the package, and train the model with seed words provided by the author[58].

### 4.2.4 Results

The experimental results are listed in Table 4.4. For RMSE, the smaller the better, while for the other two measures, the larger the better. Graph Propagation, ILDA and SATM-L do not use the overall ratings(except for training sentiment association T), so we group them together. Similarly we group Local Prediction, Global Prediction, SATM-O, SATM and SATM-W. The Overall method is a special baseline that does not do any aspect based prediction. For the LRR method, after the first step of sentence annotation, we notice that sentence-LDA fails to annotate the "h-review" of some hotel with all 7 aspects, mainly because these hotels are associated with less reviews. In this case, the LRR model will fail in the second step, so we do not include LRR in Table 4.4. Instead, we compared our method with LRR on a subset of products that comment on all aspects based on the sentence annotation. There are 1533 hotels in this subset, and the result is shown in Table 4.5. Note that our experimental results for LRR are far worse than those reported in the

[3] http://sifaka.cs.uiuc.edu/~wang296/Codes/LARA.zip

original paper[59]. We believe this maybe due to different parameter settings, or due to the choice of different reviews.

| Methods | RMSE | P@10 | $\rho_{hotel}$ |
|---|---|---|---|
| ILDA,R=2 | 1.202 | 0.30 | 0.193 |
| ILDA,R=5 | 1.096 | 0.257 | 0.222 |
| Graph Propagation | 0.718 | 0.271 | 0.442 |
| SATM-L | 0.774 | 0.443 | 0.483 |
| Local Prediction | 0.572 | 0.486 | 0.761 |
| Global Prediction | 0.625 | 0.30 | 0.778 |
| SATM-O | 0.429 | 0.80 | 0.841 |
| SATM | **0.384** | **0.814** | 0.854 |
| SATM-W | 0.407 | 0.80 | 0.842 |
| Overall | 0.415 | 0.80 | **0.863** |

Table 4.4: Experimental results except LRR

| Methods | RMSE | P@10 | $\rho_{hotel}$ |
|---|---|---|---|
| LRR | 1.018 | 0.3 | 0.404 |
| SATM | **0.373** | **0.829** | **0.849** |

Table 4.5: Experimental comparison with LRR

We observe that SATM achieves the best RMSE value, i.e., it produces the most accurate aspect rating prediction. The Overall method does better in ranking all the hotels($\rho_{hotel}$), but SATM is better at ranking top hotels($P@10$). When we compare the results of SATM with SATM-L and SATM-O, we find that the good performance of SATM is mainly due to the use of the overall rating distribution. On one hand, this is reasonable, since intuitively aspect ratings usually do not diverge too far from the overall rating, especially for hotels with higher overall ratings. As we can see from the result of Overall, the overall rating has good correlation with aspect ratings, and using overall rating only is already a strong predictor for aspect ratings. Also, in most cases, methods using overall ratings(Overall and the five methods in the middle of Table 4.4) are better than others(first four methods). On the other hand, we should not rely only on the overall rating distribution. By incorporating the sentiment lexicon, for RMSE, SATM achieves 10% improvement over SATM-O and 7% improvement than Overall. Also, the overall rating may not always be a good aspect rating predictor, depending on the dataset.

To take a closer look at cases where the overall rating is not a good aspect rating predictor, we evaluate the RMSE on different subsets of hotels. We divide the hotels into

different overall rating ranges: [1,2), [2,3), [3,4) and [4,5]. The results are shown in Table 4.6. Going from the [4,5] group to [1,2) group, the overall rating becomes less and less reliable to predict aspect ratings, and the gain of SATM increases compared to SATM-O and Overall. For a hotel with higher overall rating(good hotel), its aspect ratings are closer to the overall rating. This matches our intuition that good hotels are expected to be good on most aspects, if not on all aspects. For a hotel with average and lower overall rating, the average difference between aspect ratings and overall rating is larger. In this case, the overall rating can not tell us the whole story, which calls for aspect based prediction. Our method achieves the best RMSE gain on this group of hotels.

| Methods | [1,2) | [2,3) | [3,4) | [4-5] |
|---|---|---|---|---|
| Local Prediction | 0.789 | 0.772 | 0.621 | 0.456 |
| Global Prediction | 1.013 | 0.884 | 0.584 | 0.567 |
| SATM-O | 0.703 | 0.564 | 0.446 | 0.359 |
| SATM | **0.606** | **0.494** | **0.394** | 0.332 |
| SATM-W | 0.668 | 0.525 | 0.411 | 0.352 |
| Overall | 0.735 | 0.612 | 0.431 | **0.320** |

Table 4.6: RMSE on hotels with different overall rating ranges

We can also compare the results of the bag-of-phrase model(SATM) with bag-of-words model(SATM-W). As we can see, SATM outperforms SATM-W, especially on RMSE. One possible explanation is that the bag-of-words model fails the capture the negation of words. For example, in the sentence "the staff is not very friendly", the model may generate a positve word from a positive sentiment label. By comparison, the phrase extraction will generate a phrase $<$staff, no friendly$>$ in the preprocessing step. As we can see, the accuracy of the bag-of-phrase model relies on the preprocessing step, and the bag-of-words model needs to distinguish word types well. If the text has very informal writing style, the preprocessing step may not work well, and the bag-of-words model may achieve better performance with adequate training instances for the Max-ent classifier(See Chapter 3).

## 4.3 Qualitative analysis

To provide a qualitative analysis, we show our model is effective at aspect extraction and sentiment topic extraction.

First we list the top words for the aspect-sentiment label-word distributions. We use the aspect "room" as an example. From Figure 4.2 to 4.6 we list the 5 sentiment topics with 5 different sentiment labels. For limited space we only list the top 30 words for each distribution. We observe that, although the positive reviews are dominant in the dataset, our model manages to extract the negative sentiment topics, which usually cannot be achieved if the model is totally unsupervised. More importantly, as the sentiment label value increases, the sentiment topics express more and more positive sentiments. This means the sentiment labels and ratings are indeed aligned, so that we can use these sentiment labels to predict ratings. We also list the top words for aspects in the appendix. For comparison, we list the top words of aspect room, location, staff, service/food and business facilities for SATM and SATM-W. The topic quality from SATM and SATM-W is close in most aspects, while in certain aspect such as business facilities, the SATM-W model achieves better quality. One reason is that a sentence provides more context than a phase, so the word cooccurrence can be better learned.

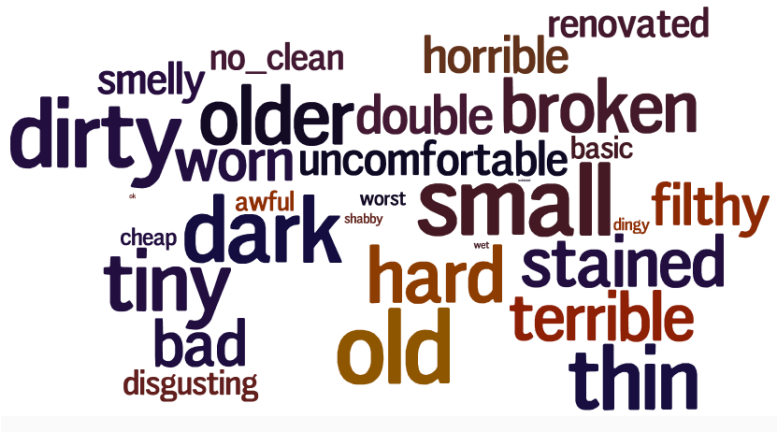


Figure 4.2: Sentiment topic about room aspect with sentiment label 1


Figure 4.3: Sentiment topic about room aspect with sentiment label 2


Figure 4.4: Sentiment topic about room aspect with sentiment label 3


Figure 4.5: Sentiment topic about room aspect with sentiment label 4


Figure 4.6: Sentiment topic about room aspect with sentiment label 5

# Chapter 5

# Conclusion and future work

In this thesis, we work on the problem of product aspect rating prediction. This task aims to extract the aspects from the product reviews, and predict the aspect ratings. It provides fine-grained sentiment analysis and helps people gain more insight into the product. Although topic modeling has been widely used to model aspect and sentiments in opinionated text, it usually cannot perform this task well due to the sentiment label alignment problem, which motivates our work.

We propose sentiment aligned topic models(SATM), incorporating two kinds of external knowledge: the overall rating distribution and a sentiment lexicon. A bag-of-phrase model is presented, and the idea is further extended to a bag-of-words model. The SATM model is effective at sentiment topic extraction, and it can align sentiment labels with ratings, so we can directly use the sentiment labels for aspect rating prediction. For quantitative analysis we evaluate the aspect rating prediction accuracy, and compare against state-of-the-art methods. Experiments demonstrate that SATM outperforms existing approaches.

We list a few directions for future work.

**Domain adaptation**. Currently, certain websites request users to explicitly provide aspect ratings for feedback, and these aspect ratings can be used in supervised learning for probabilistic models. However. these aspect ratings may be only available for a limited number of domains, such as beer, hotel, so the model may work well for these domains only. Given online reviews in a new domain, how can we transfer the knowledge from the existing domain to the new one? For example, a sentiment word with positive orientation in one domain may express negative sentiment in another domain. How to deal with the knowledge conflict is a challenge.

**Improved aspect discovery**. Aspect rating prediction can be decomposed into two tasks: aspect identification and sentiment extraction. The SATM model focuses on the second part, as it is not well explored. However, it would be interesting to combine existing methods or new methods on aspect identification with our approach for improvement. For example, aspect extraction can be supervised. Also, the number of aspects is no longer fixed. Instead, nonparametric bayesian methods can be adopted to learn the number of topics based on the data.

**Correlated and hierarchical aspects**. Aspects are hierarchical in nature, and different aspects may be correlated. Therefore, the aspect ratings can be correlated. In SATM, using the overall rating distribution for different aspect ratings can be seen as building indirect correlations, but modeling the correlations directly may be a better approach. For example, it allows different levels of correlation between different aspects.

**Fast and parallel learning**. Our method falls into the category of graphical models, and uses Gibbs Sampling for inference. On one hand, to get better performance, the model is supposed to be trained with large datasets, but as the data gets larger, the training may take a long time. How to design a fast inference method is a challenge. Alternatively, parallel training on multiple machines can be adopted for practical use.

**Weighted reviews**. In SATM, we assume that all the documents/phrases are equally important. However, this may not be true in all cases. For example, in most online review websites, users can provide feedback for "helpfulness", and reviews with high helpfulness value are supposed to be more important. In the extreme case, certain reviews may be spam or fake, so these reviews are of low importance. How to model weighted documents can be an interesting direction to work on.

**Temporal analysis**. Reviews may accumulate through time, and for different time period, the reviews may show different evaluation trends. it would be useful to build temporal models on the reviews to analyze how people express opinions at different time period, or how the service improves/worsens over time.

# Bibliography


[1] David Andrzejewski, Xiaojin Zhu, and Mark Craven. Incorporating domain knowledge into topic modeling via dirichlet forest priors. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 25--32, New York, NY, USA, 2009. ACM. 8, 10

[2] David Andrzejewski, Xiaojin Zhu, Mark Craven, and Benjamin Recht. A framework for incorporating general domain knowledge into latent dirichlet allocation using first-order logic. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, IJCAI'11, pages 1171--1177. AAAI Press, 2011. 8

[3] Sasha Blair-goldensohn, Tyler Neylon, Kerry Hannan, George A. Reis, Ryan Mcdonald, and Jeff Reynar. Building a sentiment summarizer for local service reviews. In *WWW Workshop on NLP in the Information Explosion Era*, 2008. 11

[4] David M. Blei. Probabilistic topic models. *Commun. ACM*, 55(4):77--84, 2012. 8

[5] David M. Blei and Jon D. McAuliffe. Supervised topic models. In *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007*, pages 121--128, 2007. 8, 13

[6] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993--1022, March 2003. 3, 7

[7] S.R.K. Branavan, Harr Chen, Jacob Eisenstein, and Regina Barzilay. Learning document-level semantic properties from free-text annotations. In *Proceedings of ACL-08: HLT*, pages 263--271, Columbus, Ohio, June 2008. Association for Computational Linguistics. 10

[8] Samuel Brody and Noemie Elhadad. An unsupervised aspect-sentiment model for online reviews. In *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. 3, 12, 32, 35

[9] Zhiyuan Chen, Arjun Mukherjee, and Bing Liu. Aspect extraction with automated prior knowledge learning. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 347--358, 2014. 10

[10] Zhiyuan Chen, Arjun Mukherjee, and Bing Liu. Aspect extraction with automated prior knowledge learning. In *ACL 2014*, pages 347--358. Association for Computational Linguistics, 2014.

[11] Zhiyuan Chen, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malú Castellanos, and Riddhiman Ghosh. Exploiting domain knowledge in aspect extraction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1655--1667, 2013. 3, 10

[12] Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J. Smola, Jing Jiang, and Chong Wang. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 193--202, New York, NY, USA, 2014. ACM. 10, 11

[13] Lei Fang, Minlie Huang, and Xiaoyan Zhu. Exploring weakly supervised latent sentiment explanations for aspect-level review analysis. In *Proceedings of the 22Nd ACM International Conference on Conference on Information & Knowledge Management*, CIKM '13, pages 1057--1066, New York, NY, USA, 2013. ACM.

[14] Geli Fei, Zhiyuan Chen, and Bing Liu. Review topic discovery with phrases using the pólya urn model. In *COLING 2014, 25th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, August 23-29, 2014, Dublin, Ireland*, pages 667--676, 2014. 11

[15] Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl. 1):5228--5235, April 2004. 19

[16] Zhen Hai, Gao Cong, Kuiyu Chang, Wenting Liu, and Peng Cheng. Coarse-to-fine review selection via supervised joint aspect and sentiment model. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, pages 617--626, New York, NY, USA, 2014. ACM. 11

[17] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, pages 50--57, New York, NY, USA, 1999. ACM. 3, 7

[18] 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. 1, 11

[19] Jagadeesh Jagarlamudi, Hal Daumé, III, and Raghavendra Udupa. Incorporating lexical priors into topic models. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '12, pages 204--213, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics. 34

[20] 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. 3, 4, 9, 25, 33

[21] Suin Kim, Jianwen Zhang, Zheng Chen, Alice H. Oh, and Shixia Liu. A hierarchical aspect-sentiment model for online reviews. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence, July 14-18, 2013, Bellevue, Washington, USA.*, 2013. 3, 4, 11

[22] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282--289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. 12

[23] 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 2011, April 28-30, 2011, Mesa, Arizona, USA*, pages 498--509, 2011. 3, 4, 10, 13

[24] Angeliki Lazaridou, Ivan Titov, and Caroline Sporleder. A bayesian model for joint unsupervised induction of sentiment, aspect and discourse representations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 1630--1639, 2013. 3

[25] Chengtao Li, Jianwen Zhang, Jian-Tao Sun, and Zheng Chen. Sentiment topic model with decomposed prior. In *SIAM International Conference on Data Mining (SDM'13)*. Society for Industrial and Applied Mathematics, 2013. 3, 4

[26] 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, Atlanta, Georgia, USA, July 11-15, 2010*, 2010. 3, 4, 10

[27] Kar Wai Lim and Wray Buntine. Twitter opinion topic model: Extracting product opinions from tweets by leveraging hashtags and sentiment lexicon. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, CIKM '14, pages 1319--1328, New York, NY, USA, 2014. ACM. 10, 11

[28] 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. 3, 4, 9, 17

[29] Guang Ling, Michael R. Lyu, and Irwin King. Ratings meet reviews, a combined approach to recommend. In *Proceedings of the 8th ACM Conference on Recommender Systems*, RecSys '14, pages 105--112, New York, NY, USA, 2014. ACM. 11

[30] Bing Liu. *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2012. 1, 3, 9, 11, 12

[31] Chong Long, Jie Zhang, Minlie Huang, Xiaoyan Zhu, Ming Li, and Bin Ma. Estimating feature ratings through an effective review selection approach. *Knowl. Inf. Syst.*, 38(2):419--446, 2014.

[32] 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. 11, 12, 15, 32, 35

[33] Julian McAuley, Jure Leskovec, and Dan Jurafsky. Learning attitudes and attributes from multi-aspect reviews. In *Proceedings of the 2012 IEEE 12th International Conference on Data Mining*, ICDM '12, pages 1020--1025, Washington, DC, USA, 2012. IEEE Computer Society. 13

[34] 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. 3

[35] David M. Mimno and Andrew McCallum. Topic models conditioned on arbitrary features with dirichlet-multinomial regression. In *the Conference on Uncertainty in Artificial Intelligence*, pages 411--418, 2008. 9, 18, 36

[36] 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. 3, 11, 13, 15, 33

[37] Samaneh Moghaddam and Martin Ester. Aspect-based opinion mining from product reviews. In *The 35th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '12, Portland, OR, USA, August 12-16, 2012*, page 1184, 2012. 1

[38] 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. 11, 12, 15, 30

[39] Samaneh Moghaddam and Martin Ester. The flda model for aspect-based opinion mining: Addressing the cold start problem. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 909--918, 2013. 3, 11, 13, 15

[40] 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. 3, 10, 28, 29, 34

[41] S. Mukherjee et al. Joint author sentiment topic model. In *In Proc. of SIAM International Conference on Data Mining (SDM 2014)*. SIAM, 2014. 10

[42] Nikolaos Pappas and Andrei Popescu-Belis. Explaining the stars: Weighted multiple-instance learning for aspect-based sentiment analysis. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 455--466, Doha, Qatar, October 2014. Association for Computational Linguistics. 13

[43] 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. 11

[44] Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. Opinion word expansion and target extraction through double propagation. *Comput. Linguist.*, 37(1):9--27, March 2011. 12

[45] Lawrence R. Rabiner. Readings in speech recognition. chapter A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, pages 267--296. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1990. 12

[46] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pages 248--256, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. 8

[47] Michal Rosen-Zvi, Thomas Griffiths, Mark Steyvers, and Padhraic Smyth. The author-topic model for authors and documents. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, UAI '04, pages 487--494, Arlington, Virginia, United States, 2004. AUAI Press. 8

[48] Christina Sauper and Regina Barzilay. Automatic aggregation by joint modeling of aspects and values. *J. Artif. Int. Res.*, 46(1):89--127, January 2013. 10, 13

[49] Christina Sauper, Aria Haghighi, and Regina Barzilay. Incorporating content structure into text analysis applications. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 377--387, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[50] 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. 3, 4, 10, 13

[51] Benjamin Snyder and Regina Barzilay. Multiple aspect ranking using the good grief algorithm. In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, April 22-27, 2007, Rochester, New York, USA*, pages 300--307, 2007. 13

[52] Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. Lexicon-based methods for sentiment analysis. *Comput. Linguist.*, 37(2):267--307, June 2011. 5, 15, 34, 36

[53] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566--1581, 2006.

[54] 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. 3, 9

[55] Ivan Titov and Ryan T. McDonald. A joint model of text and aspect ratings for sentiment summarization. In *ACL 2008, Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, June 15-20, 2008, Columbus, Ohio, USA*, pages 308--316, 2008. 2, 13

[56] 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. 5

[57] Hanna M. Wallach, David M. Mimno, and Andrew McCallum. Rethinking LDA: why priors matter. In *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada.*, pages 1973--1981, 2009.

[58] Hongning Wang, Yue Lu, and Chengxiang Zhai. Latent aspect rating analysis on review text data: A rating regression approach. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, pages 783--792, New York, NY, USA, 2010. ACM. 4, 12, 13, 30, 31, 32, 33, 34, 36

[59] 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, New York, NY, USA, 2011. ACM. 4, 12, 13, 30, 33, 37

[60] 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. 8

[61] Zhongwu Zhai, Bing Liu, Hua Xu, and Peifa Jia. Clustering product features for opinion mining. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 347--354, New York, NY, USA, 2011. ACM. 11

[62] 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. 10, 29

[63] Xiaojin Zhu and Zoubin Ghahramani. Learning from labeled and unlabeled data with label propagation. Technical report, Technical Report CMU-CALD-02-107, Carnegie Mellon University, 2002. 12

[64] 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. 12

# Appendix A

# Aspect topic examples


Figure A.1: The room aspect-word distribution for SATM-W


Figure A.2: The room aspect-word distribution for SATM


Figure A.3: The location aspect-word distribution for SATM-W


Figure A.4: The location aspect-word distribution for SATM


Figure A.5: The staff aspect-word distribution for SATM-W


Figure A.6: The staff aspect-word distribution for SATM

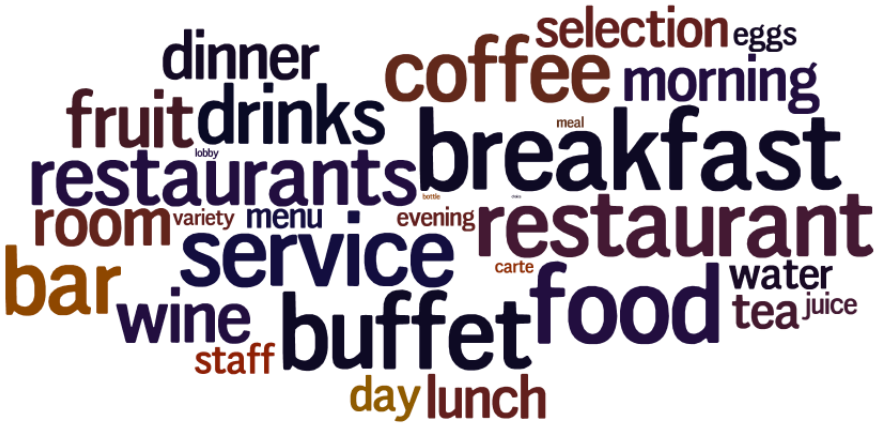


Figure A.7: The service/food aspect-word distribution for SATM-W


Figure A.8: The service/food aspect-word distribution for SATM


Figure A.9: The business/facilities aspect-word distribution for SATM-W

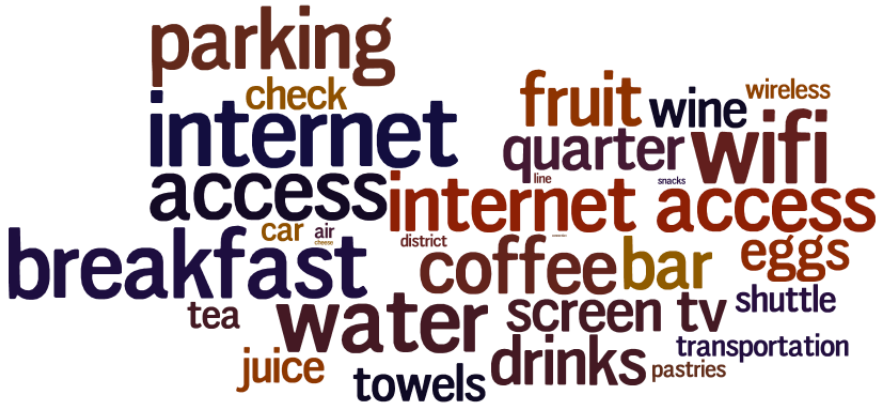


Figure A.10: The business/facilities aspect-word distribution for SATM