Semi-supervised Extraction of Entity Aspects

Using Topic Models

Mehrbod Sharifi

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave. Pittsburgh, PA 15213
www.lti.cs.cmu.edu

Thesis Committee:

Eric Nyberg, Chair

Jaime Carbonell

William Cohen

Submitted in partial fulfillment of the requirements
for the degree of Masters of Science

Abstract

Information extraction techniques (such as Named Entity Recognition) have long been used to extract useful pieces of information from text. The types of information to be extracted are generally fixed and well defined (e.g., names of people, organizations, etc.). However in some cases, the user goal is more abstract and information types cannot be narrowly defined. For example, a reader of online user reviews typically has the goal of making a good choice and is interested to learn about the different aspects or attributes that people have mentioned for an entity (e.g., quality of service for a restaurant or battery life of a digital camera). Some of these aspects may be known by the reader and some others may need to be discovered from the inherent text structure in a large collection. Even for the known aspects (such as “service” for a restaurant), the challenge is to recognize various expressions (e.g., “long wait” or “friendly waiter”).

In this thesis, we model the entity aspects as topics with identifiable word distributions across documents. We review several probabilistic graphical models (such as Latent Dirichlet Allocation) and propose a new model which can operate in a semi-supervised setting. We provide empirical evaluation for the success of this model in biasing the natural topic distribution toward entity aspects in user reviews.

Acknowledgements

I would like to thank my advisor, Eric Nyberg, who patiently guided me through the process of putting together my scattered thoughts into this thesis. I am very grateful to my committee members William Cohen and Jaime Carbonell who helped me shape and refine the ideas presented here with exciting discussions. My interest in graphical models significantly increased while taking the Graphical Model course with Carlos Guestrin and his amazing teaching style. Amr Ahmad, both when he was the TA of this course and afterward, played an instrumental role in completion of thesis by helping me get through the complexities of designing and running a Gibbs Sampling procedure for the new model.

I am dedicating this thesis to my parents who are the primary reason for anything that I achieve in my life and to my wife for her infinite and genuine support and patience while I was working on this thesis.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

Information extraction is typically performed in the following setting: given an information type in a type system, specify all segments of text which are instances of this type. A type system is analogous to a database schema where we define the semantics for each field. For example, in a typical Named Entity Recognition task, we are looking for segments of text which are instances of the types: person, organization or location. The types of desired information are fixed and while the expression of this information can vary substantially in text, it is often the case that many contextual clues and patterns that can be learned for these extractions remain the same. In some cases (e.g., web pages), it is even possible to take advantage of clues other than textual contents for the extraction (e.g., formatting differences for the names of the person on their personal web page vs. other contents).

For some applications however, we need a more flexible approach to extract information we are looking for because the definition of the types is more abstract and driven by inherent structure in the document collection and specific user goal. The information extraction model we are interested should discover the types and the instances simultaneously.

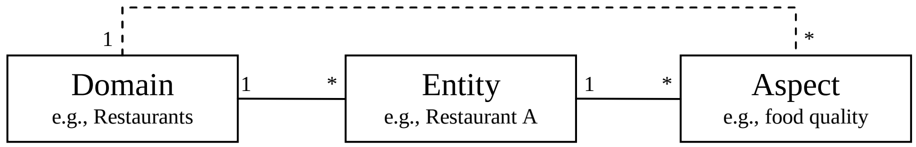


Figure 1-1: Terminology

1.2 Problem Statement

We focus on the special case of the problem described above where we are interested to extract the different aspects of an entity. We are looking for a concise answer to the questions of "what do I need to know about entity X in domain Y". Our terminology is summarized in Figure 1-1. A domain is a category for a series of entities (or items), e.g., Digital Cameras. An entity refers to a concept (person, location, service, etc.) that the document is discussing or describing e.g., a specific brand of digital camera. The aspects are the properties or attributes of the entity that are mentioned about the entity and are often of interest to the user. In the case of a digital camera as entity, the picture quality or the battery life are the examples of the aspects. The dotted line in the Figure 1-1 refers to the fact that we have some *a priori* knowledge of domain aspects. The task is to recognize the mentions of each aspect and also to mark the span of text where the aspect is mentioned. For example, consider the following review:

> The sushi is very good and it's not expensive at all. You have to try their special rolls. The place is not fancy at all but staffs are friendly.

The review is about a Sushi restaurant (the entity) and refers to the following aspects: "food", "price", "atmosphere" and "service". We also are interested to mark the spans as follows:

“Food”: The sushi is very good and
“Price”: it's not expensive at all.
“Food”: You have to try their special rolls.
“Atmosphere”: The place is not fancy at all but
“Service”: staffs are friendly.

Appendix A has more examples from our annotated dataset.

On the surface, this seems like a typical text classification task but some characteristics of the task make it hard for the conventional classification method:

1. The list of aspects is open-ended and needs to be discovered from the corpus. As was the case, in the example above, the aspect labels often are not explicitly mentioned in the text. This is similar to problem of finding cluster labeling in unsupervised learning.
2. There is considerable variability in the contexts of the expressions we would like to extract. For example, “long wait” and “friendly waiter” both are related to the service quality while they appear in different contexts.
3. There is an issue with “level of granularity”: how broad or specific each aspect should be. For example, should the *atmosphere* be broken down further to décor, ambience, etc.

The good news is that there is often certain level of flexibility on the part of the user as to what these aspects “can” be. Therefore, we may be able to take advantage of the inherent structure of corpus to suggest the aspects. Also, often an initial set of the aspects may be known and the goal is to discover more. We can also obtain the user’s goal by requesting a few instances to be labeled.

There is more discussion in literature about the differences between this task and the traditional classification approaches (Hu & Liu., 2004; Branavan et al, 2008). In this thesis, we review several existing solutions and then present ours which addresses this task directly. Our model takes advantage of minimal supervision (i.e., annotations of entity aspect) and then generalizes to discover the topics (i.e., distributions of words) such that each topic closely corresponds to an entity aspect.

We make the following assumptions about the input data:

1. The dataset consists of multiple documents that are written to explain aspects of one known entity. This is a very important assumption because we exploit the comparative statistic of words across documents to infer attributes. It is possible that some parts of the documents mention other entities as well but most of the document content should be directly or indirectly about the entity in question.
2. The entity has a finite set of sufficiently distinct attributes.
3. Since we are operating in a semi-supervised setting, some of the documents are labeled with the desired aspects of the entity. This number can be very small.

This task is fairly general. Broadcast news, blogs and many other sources of text are also dealing with reporting various aspects of entities (e.g., an aspect of a news event about a company could be “legal trouble”). We expect the task to be more challenging in those broader applications as the number aspects can be very high and sometimes more confusing mix of information even for human. In this thesis, we did not have sufficient time to annotate and perform experiments on multiple datasets and therefore the approach is tested only on the restaurant reviews domain.

1.3 Contributions

In this thesis, we provide a review of existing approaches to problems similar to what we outlined in the previous section. We will use this knowledge to design our approach by extending the basic LDA model into a Semi-supervised LDA model (SS-LDA). We hypothesize that SS-

LDA performs better than our baseline LDA in discovering the aspects and specifying the corresponding spans of text. Performance is measured by precision/recall on a manually annotated test set which is created as part of this thesis. We show that this performance improvement is related to the fact that the model is successfully using the side information (i.e., small number of labeled instances) to bias the topic distributions toward the user goal. We measure the distribution distances and also observe the change in perplexity of held out set.

Chapter 2 provides the literature review of related works. Chapter 3 has the complete details of the new approach, experiments conducted and evaluations. In chapter 4, we discuss possible future work in this area.

We will make the dataset of this thesis (Appendix A) and the software from this thesis called STAT (Appendix B) publicly available.

Chapter 2

Related Work

In this chapter, we will provide a literature review for related work in this area. For two reasons, the list of related works is long: one reason is that this task is at the intersection of several important and active machine learning and NLP research areas therefore many different approaches can be adapted to become relevant. The other important reason is that our main approach of using graphical models has gained tremendous interest in recent years due to successful application of these methods. We have made an attempt to organize some of related literature with respect to their relevance to this task. We hope the result of this literature review to help anyone who would like to design a new method for a different task based on what is known about previous methods[1].

2.1 Summarizing Product Reviews

The task of extracting entity aspects was discussed in some of the previous literatures as summarization of product reviews. The methods are typically large systems in which various steps are performed in a “pipeline” setting to extract the mentions of the attributes. We will only

[1] We have not implemented or experimented with any of these methods other than the basic LDA due to time restriction in adapting them to our task. The purpose of mentioning them is to understand their model design.

describe two of such systems and provide sufficient details for their approaches which used to be popular in this area but are fundamentally different than our approach.

Hu and Liu (2004) introduced the frequently used (and the only) dataset for this summarization task which is composed of a set of consumer electronic user reviews. We will describe the dataset in 3.2.1 and explain why we could not use it for our task. Their system, called FBS (Feature-based summarization), performs both the extraction of the product attributes (which we call entity aspects) and also the sentiment analysis for these attributes (i.e., whether the user liked them or not). For the extraction part, they use association mining on noun phrases (hence POS tagged and chunk parsed) followed by several heuristic pruning steps (compactness pruning to remove meaningless phrases and redundancy pruning to remove the subset features). Further in the process, after finding the opinion phrases, they extract more features which are infrequent by looking at the sentences that have opinion phrases but no features from the first step. Extraction performance is evaluated against the manual annotation in the dataset and is reported for each step and each product group. The overall precision is 72% and recall 80% which outperforms their baseline, FASTR term extraction system: precision 3% and recall 16%

Popescu and Etzioni (2005) perform experiments on the same dataset using their OPINE system and improve the result from Hu and Liu (2004). Their system first performs a parse of the input, resolves pronouns and takes all the noun phrases (frequency above a tuned threshold). They use extraction patterns based on dependency parse to generate a set of discriminator phrases (e.g., “scanner has a”). Then they compute the point-wise mutual information (PMI) between the phrases in these sets and consider a subset of noun phrases with PMI above the threshold as product features. Their system improves the previous extraction precision by 22% with loss of 3% in recall.

While these systems could be tuned to perform relatively well for a specific domain, a lot of engineering is needed for various pre- and post-processing steps. As a result, they are hard to be reproduced and their performance cannot be easily understood. In recent year, methods similar to ours gained more popularity because they are simpler to design and understand. We model the entire problem and then use standard statistical techniques to train and apply the model. Furthermore, the performance of these newer methods is often comparable in various domains or even across different languages with small modifications.

2.2 Hierarchical Bayesian Graphical Models

Probabilistic graphical models have gained tremendous attention in machine learning and NLP research area in recent years. These models provide a concise and intuitive representation of part of the world in terms of random variables (i.e., concepts in real world such as reading from a temperature sensor). They express the relations between the variables and allow inference in presence any amount of observed information for those variables. Graphical models have been studied in machine learning and statistics for many years and there are well established methods for learning their structures and parameters.

In this section, we provide a comprehensive literature review of the work in this area which is related to our task and will help us design our new approach.

2.2.1 Notation

We will briefly explain the graphical model notation. A graphical model is a graphical representation of how a joint probability distribution over certain number of variables factorizes by encoding independence assumptions between variables. Circles are used for random variables whose probability distribution parameters (typically represented with Greek alphabet). The "hierarchical" aspect of some graphical models is when the variables are stacked in sequence and their relation is in a hierarchical form. Plate notation (squares around a subset of circles) is used represent replication of the parameters to the number that is in the lower right corner of the square. Arrows represent possible dependency between variables. Shaded circles are the variables that are observed in the data. Figure 2-1 shows a few simple examples of graphical models which we will explain in the next section.

2.2.2 Dimensionality Reduction

This topic is not directly related to graphical models but we believe it is one of the practical reasons for the adoption these methods in NLP tasks. In modeling text, we typically consider words as features and for example, their frequency in document as feature value. Since the size of vocabulary is often large, most techniques on text suffer from the curse of dimensionality and hence we always need methods to reduce the dimensionality. Sometimes this is accomplished by simple "feature selection" such as eliminating stop words or truncating the vocabulary based on the minimum term document frequency. However, in some applications we would like to find low dimensional representation of documents which also encode the semantic properties as well. For example, in information retrieval (IR), we need a low dimensional vector representation such that the documents that are semantically similar to have similar vector representation even if they are syntactically different (i.e., do not share similar words)

Latent Semantic Indexing or LSI (Deerwester et al, 1990) was one of the methods designed to reduce dimensionality with this goal. The critical step in LSI is the Singular Value Decomposition (SVD), closely related to Principle Component Analysis (PCA), which is an algebraic (and computationally intensive) procedure that decomposes the document-term frequency matrix A into three components: $A = USV^T$. U matrix is the unit vectors[2] for terms and V matrix is the unit vectors for documents and diagonal, sorted S matrix is the singular values. To perform dimensionality reduction, the S matrix was truncated (i.e., values smaller than a threshold were made zero) and then SV^T is the projections of the documents into the lower dimensions equal to the number of non-zero element left in the diagonal of S. This method minimizes the sum of squared difference (Euclidean distance) between the projected vectors and the original vectors.

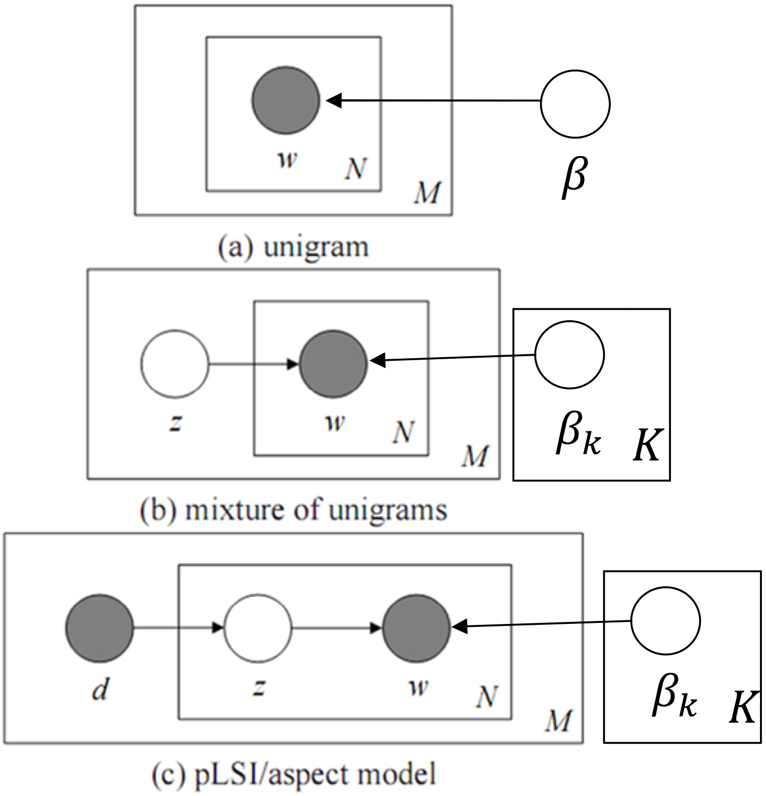


Figure 2-1 Document Models (Blei et al, 2003)

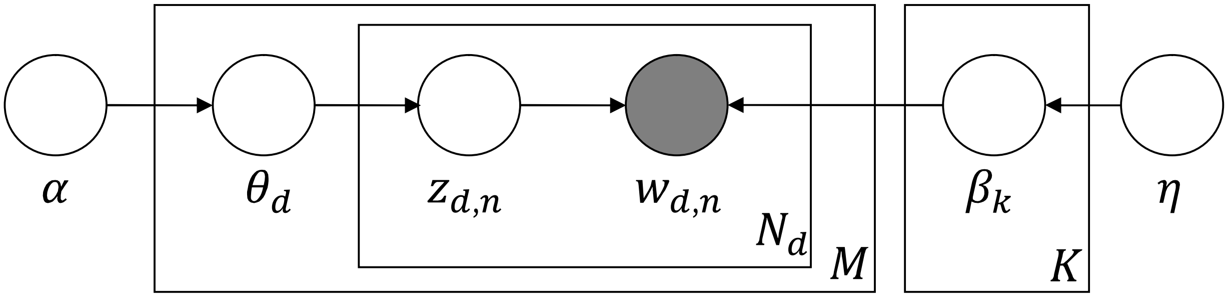
[2] These unit vectors are the basis vectors which are spanning the projected space for terms or documents

This method is unsupervised and only uses the co-occurrence frequency counts provided in the document-term matrix. LSI has been used in many applications other than IR (e.g., in Word Sense Discrimination (Levin & Sharifi, 2006)). This approach is still one of the simplest dimensionality reduction approaches.

Clustering in general can be considered dimensionality reduction approaches: a soft assignment of an instance to a set of clusters is a low dimensional vector representation of the instance.

A simple form of clustering is to model a document collection as mixture of multinomial distributions over words (Figure 2-1b) and then discover the components using the EM algorithm. This approach has been very well studied (Nigam et al, 2000). There is a problem with identifiability of these components. For discussion on this and also similar algorithms see the literature survey on semi-supervised learning by Zhu (2005).

Probabilistic LSI or pLSI (Hofmann, 1999) is a method that combines the idea of LSI and EM for identifying the probability distribution of mixtures (Figure 2-1c). This model takes into consideration that there are multiple topics for documents but does not explicitly model how the mixture of topic is determined (variable d is a document index; to find a topic mixture a new test instance heuristics was used in the paper). This issue is addressed in LDA as discussed in details in Blei et al (2003).



Hyper parameters

- α : Topic mixture prior $_{1 \times K}$ (Number of topics)
- η : Topic word distribution prior (Smoothing) $_{1 \times V}$ (Vocabulary size)

Generative Process

1. For each topic: $k = 1, 2, \ldots, K$

   $\beta_k \sim Dirichlet(\eta)$: Choose a topic word distribution $_{1 \times V}$
2. For each document: $d = 1, \ldots, M$
   a. $\theta_d \sim Dirichlet(\alpha)$: Choose a topic mixture $_{1 \times K}$
   b. For each word: $n = 1, \ldots, N_d$
      i. $k = z_{d,n} \sim Multinomial(\theta_d)$: Choose a topic $_{1 \times 1}$
      ii. $w_{d,n} \sim Multinomial(\beta_k)$: Choose a word $_{1 \times 1}$

Figure 2-2: Latent Dirichlet Allocation (LDA)

2.2.3 Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation or LDA (Blei et al, 2003) is a hierarchical Bayesian model that capture the thematic information in a document collection as topics and addresses the problem in pLSI by using a Dirichlet prior on topic mixture. Topics are distributions over words and the model learns them by taking into account both document level and corpus level term frequency information.

Figure 2-2 shows the graphical model for LDA and its generative process which describes how the model assumes the documents are generated. On the right side of the graphical model, LDA learns the topic distribution based on the information in the entire corpus (across document or global). On the left side, the parameters for topic mixture is consider the information from within the document by assigning the topics to each word (through z).

Learning parameters in some graphical models require inference and exact inference is intractable in LDA due to the coupling between the topics z and β (Blei et al, 2003). Therefore approximate techniques need be used (which are NP-complete). The most common approaches that are used for approximate inference are the following:

1. **Gibbs Sampling**: It is a special case of Metropolis-Hasting algorithm and therefore a Markov Chain Monte Carlo (MCMC) method. Since the conditional between variables are known, we can sample each variable separately given all other variables. These samples form a Markov Chain and it is shown that the stationary distribution of the chain converges to the true posterior distribution.
2. **Variational Method**: Using Jensen's inequality, we can create a series of model where the coupling between variables is removed. The models are specified by the variational parameters and every model provides a lower bound on the log-likelihood of the original model. Variational EM is then used to learn those parameters from the data: in the E step, variation parameters are optimized by inference in variational model (which is tractable) and then in M step, the model parameters are optimized to maximize the lower bound on the log-likelihood (typically using a gradient-based approach).
3. **Expectation Propagation**: A form of message-passing algorithm, which is used less frequently for LDA that the other two approaches (Minka & Lafferty, 2002).

Finding posterior distribution for of all models discussed here is a non-convex problem and has many local optima. Many experimental tricks (such as special initialization, multiple restarts, etc.) are needed to avoid bad local optima.

Table 2-1 summarizes some practical differences between the two most popular methods. We have implemented both variational method and Gibbs sampling in this thesis, however we chose the Gibbs sampling to report the results.

	Pros	Cons
Gibbs Sampling	• Often simple to derive • More accurate approximation • Many existing software tools	• Convergence cannot be observed • Stochastic rate of convergence (i.e., slower) • Needs conjugacy between distributions to be efficient
Variational Method	• Standard optimization techniques • Often converge quickly	• Sometimes difficult to derive • Less accurate approximation

Table 2-1: Variational Method vs. Gibbs Sampling

In recent years, LDA has become the basis of many other more complicated generative models and we review some of the ones more relevant to our work in the next sections.

We group the previous models by how they extend the basic LDA. The first group of models is linking the topics so they represent a more realistic assumption about the topics exchangeability than LDA. This is often through introducing different independence assumptions (adding/removing arrows) and/or changing priors on topic mixture (θ) or topic assignments (z). The second group of models is adding more variables to incorporate more information in the model (e.g., labels, authors).

2.2.4 Extending LDA by Linking the Topics

2.2.4.1 Correlated Topic Models (CTM)

LDA uses the Dirichlet prior on the topic mixtures (θ) which makes the assumption that topics are exchangeable and independent. This however, often is not the case and Blei & Lafferty

(2007) address this problem by considering the correlation between topics by using the log-normal prior (Σ and μ in Figure 2-3). This prior is no longer conjugate to multinomial distribution of the topic assignment and they derive an approximate inference based on variational methods. They tested the models on a collection of *science* articles and showed an improvement in the perplexity.

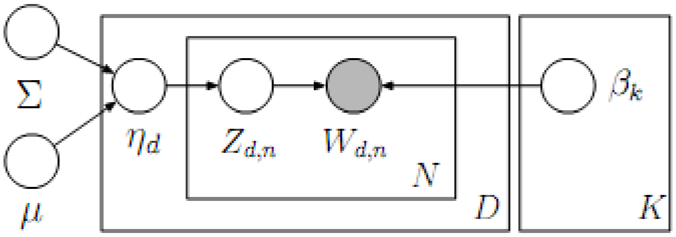


Figure 2-3 Correlated Topic Models (CTM) - (Blei & Lafferty, 2007)

Considering correlation between aspect in our task is important because it is obvious that not all aspects are independent, especially since we let the aspect be discovered from the corpus. However, in the simpler setting that we are testing our method with (i.e., using the fix set of six aspects) this correlation may not important.

2.2.4.2 Dynamic Topic Models (DTM)

To model the topics evolution over time (e.g., scientific topics in papers), Blei & Lafferty (2006) proposed the model shown in Figure 2-4. Each vertical segment is a regular LDA designated to certain time span and the parameters of the LDA (α and β) are dependent of each other as shown in the graphical model.

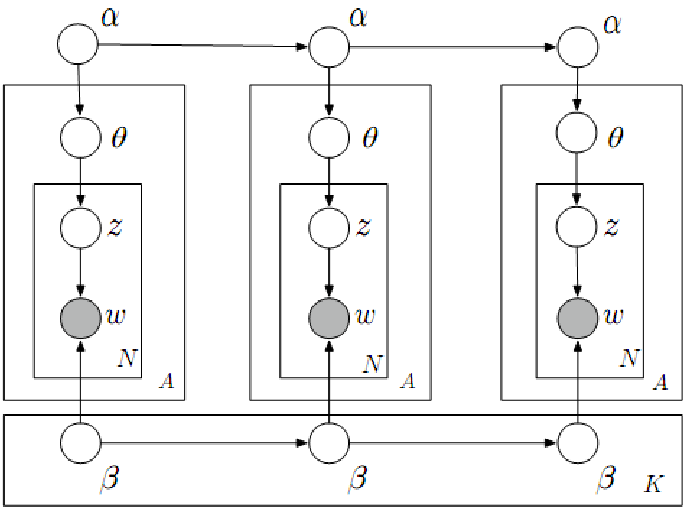


Figure 2-4: Dynamic Topic Model (DTM) (Blei & Lafferty, 2006)

This model allows capture transition between topics distributions. The model can be helpful when analyzing the trends in product features (e.g., HD feature for TVs became important at some point in time few years ago) or how the aspect of a service improves or degrades over time. For our problem we are possibility more interested in the transition between the topics within the document (next two sections) as opposed to transition between the whole topic word distributions.

2.2.4.3 HMM-LDA

Griffiths et al (2005) introduce an approach to embed HMM in LDA. As can be seen from Figure 2-5, the topic portion is a regular LDA with topic assignments ***z*** and the bottom portion is an HMM with hidden states ***s***. This will let the model to learn the semantic part of the data jointly with the syntactic part. Their results show the capability of the model in separating these two aspects of the data by grouping the content words and functions word in different topics and capture the transition between them. This is not what we are interested but it is interesting contrast this effect with the alternative approach in the next section. The interesting part for us in this model is how the topic assignments z is affected when the exchangeability assumption is removed by imposing the HMM: the topics are biased toward what the HMM needs for the hidden states.

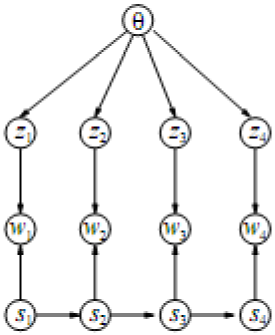


Figure 2-5: HMM-LDA (Griffiths et al, 2005)

2.2.4.4 Hidden Topic Markov Model (HTMM)

In Gruber et al (2007) a different combination of HMM and LDA is presented. As it can be seen from Figure 2-6, the transition model is defined on topics instead of words (the left two figures are equivalent and just expanding of the plate notation). HMM part of the model is the combination of ϵ, ψ and observed variables z (for when HMM is trained) and each sentence is considered a document. In contrast to HMM-LDA, this forces the model to learn topics that follow a sequence and intuitively it is a more sound assumption than what the LDA makes.

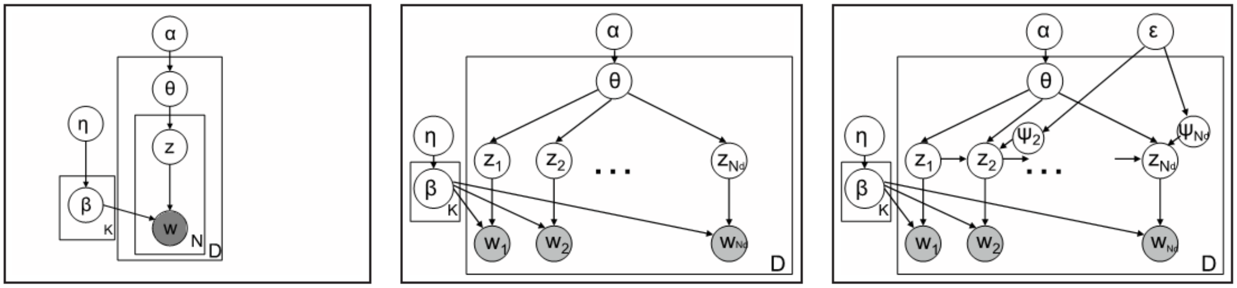


Figure 2-6 Hidden Topic Markov Model (HTMM) (Gruber et al, 2007)

The model is tested on the NIPS papers dataset and they showed an example that in LDA the word “support” was assigned to the same topic when it was part of “Support vector machines” or in the acknowledgement section of the papers (which really means the support) and the HTMM can correctly recognized this semantic difference. This is very useful for our purpose because it is known that people tend to follow a sequence when discussing aspects (e.g., typically, overall comments at the beginning or the end of the review). Also, this can help with the issue that LDA typically creates topics that are not cohesive (see section 4.1 for our suggestion of how to address this issue). On the other hand, this model may be too restrictive as people tend to mix topics in the same sentence (see the example in section 1.2). It is still interesting to investigate in future work how this model would perform in our task (when adapted to use the supervision from the labeled instances).

2.2.4.5 Markov Topic Models (MTM)

Wang et al (2009) modelled the relation between word topic distribution (β) by combining them in a Gaussian Markov random field which is an undirected graphical model. This model is shown in Figure 2-7.

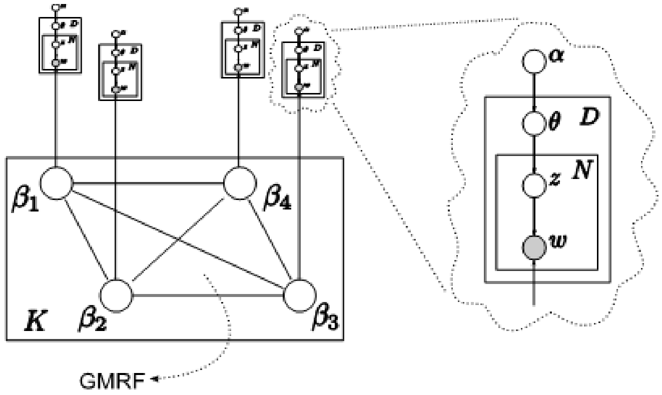


Figure 2-7 Markov Topic Models (MTM) (Wang et al, 2009)

This model is similar to DTM for but it model does not require a sequence of the topics (fewer assumptions). This allows the model to identify natural separation of the corpus and train individual an LDA for each. In the paper, they combined multiple corpora and the model separated them and also provided information about the correlation between them. The training of this model is quite complicated and may not be very helpful for our problem since (as mention in DTM) we do not have any natural segmentation of the corpus.

2.2.4.6 Syntactic Topic Models (STM)

Boyd-Graber & Blei (2009) proposed STM where the syntactic structure (i.e., parse) of a sentence can influence the topic assignment in LDA. This moves the exchangeability assumption of the LDA from word to sentences (similar to the HMM model) and forces a parse structure to be followed. In Figure 2-8, the plate with ∞ represent the non-parametric aspect for word topic distribution τ and sentence transition model π which means that the number of those parameter can grow with the data as needed. β is the prior for the topic proportion in the document (θ as in LDA) but it also control the transition model of topics within sentences (π).

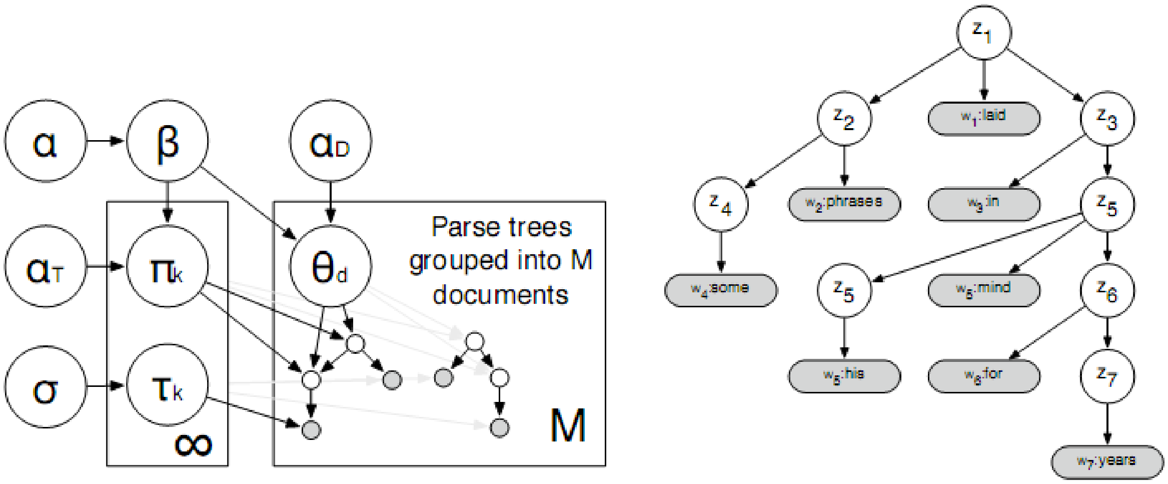


Figure 2-8 Syntactic Topic Models (Boyd-Graber & Blei, 2009)

This model is not directly relevant to our task (except if used as is and evaluated as regular LDA was evaluated) but we have mentioned to show another possibility of biasing the topic assignments.

2.2.5 Extending LDA by Modeling Further Information

2.2.5.1 Using Features

We can run the LDA and consider the observables to be features extracted from text instead of words (e.g., word n-grams). Haghighi & Klein (2007) applied this method to the task of entity co-reference resolution task. Figure 2-9 the features that are added between the word observations and regular LDA topics (e.g., G2 is a gender feature) to bias the topic distribution and ultimately improve the prediction performance.

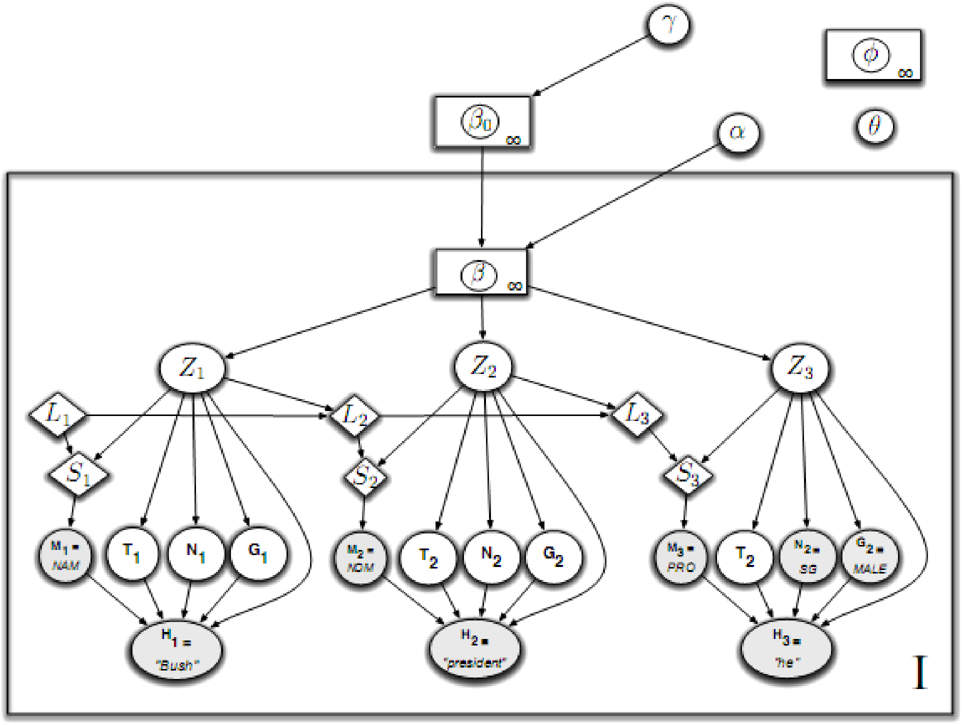
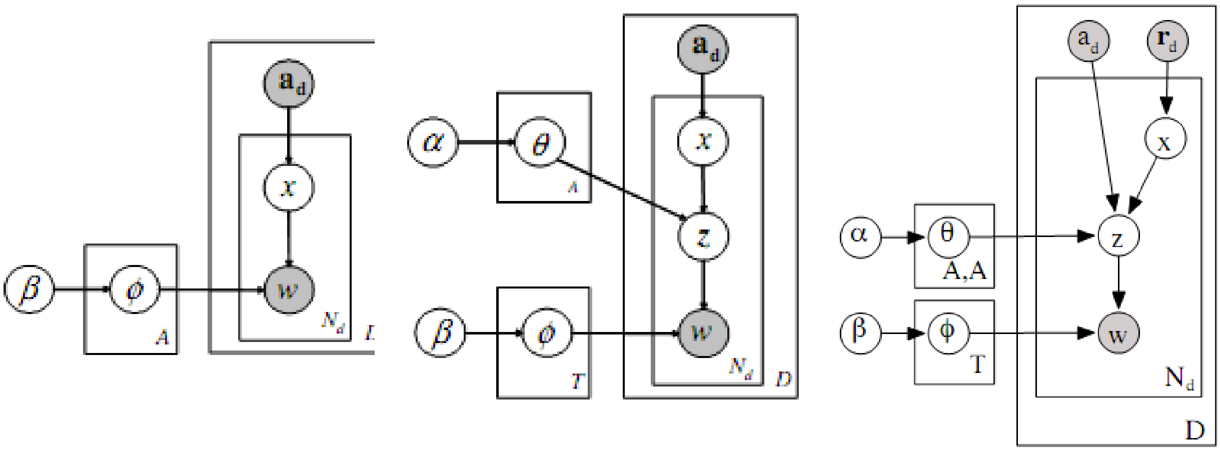


Figure 2-9 Using Features for Co-reference Resolution Task (Haghighi & Klein, 2007)

2.2.5.2 Document Authors

Figure 2-10a shows a model for the documents and the authors. Authors a_d are observed and each author has a language model. To combine this model with LDA (Figure 2-10b), we can let the author assignment x to influence the topic assignment z. The model learns the topics for each author θ which is selected by x and then based on these a topic z is chosen which selects the corpus level language mode that generates the word. Figure 2-10c is a model which incorporates the sender and recipients of email documents (as two sets of authors) by having another variable that influences z. We bias topics by changing what it is generating instead of changing its prior. The rationale behind this approach is explained in the next sections.

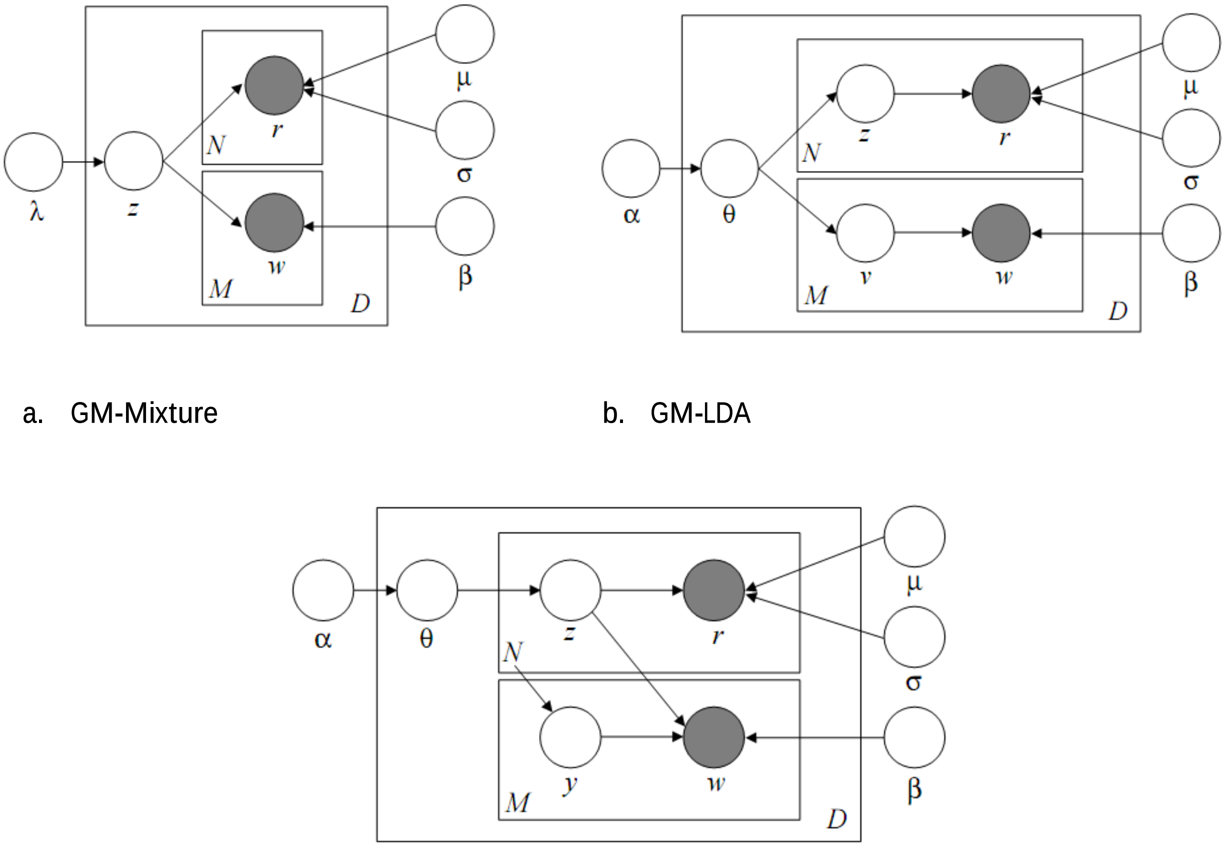


a. Author Model (AM)

b. Author Topic Model (ATM) (Rosen-Zvi et al, 2004)

c. Author Topic Recipient Model (ARTM) (Mccallum et al, 2005)

Figure 2-10 Adding Authors and Recipients to LDA



a. GM-Mixture

b. GM-LDA

c. Correspondence LDA (CorLDA)

Figure 2-11 Correspondence LDA (Blei & Jordan, 2003)

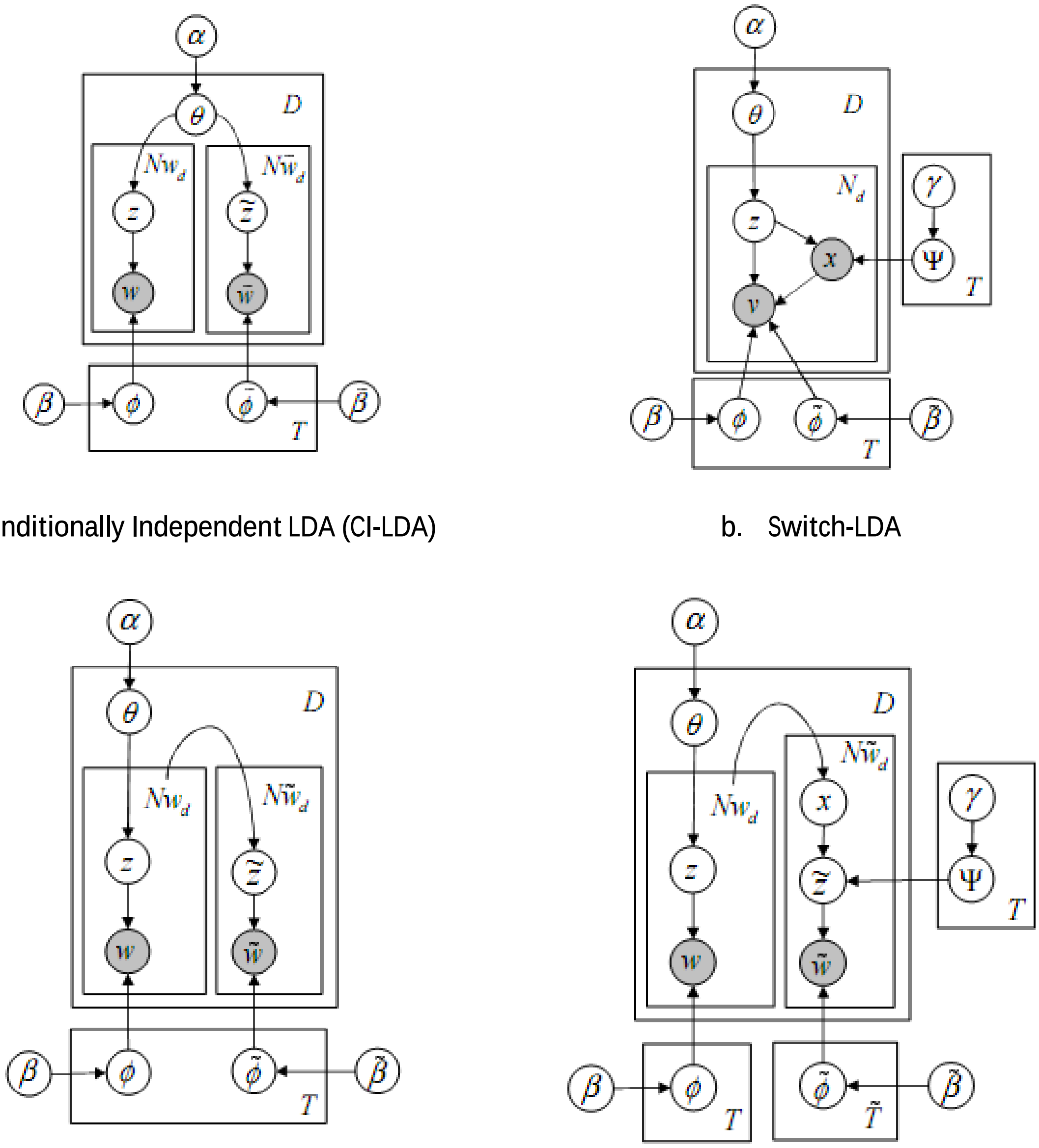
2.2.5.3 Tagged Images (Correspondence)

Blei & Jordan (2003) explain several options to create a joint model between image regions (r) and textual annotation tags for images (w). Figure 2-11a models the problem as a mixture of Gaussian where each component is indexed by z. Their results show that this model overfits. Figure 2-11b uses two parallel LDA that are sharing the same mixture parameter for topic assignment. While the perplexity of this option was comparable with the CorLDA, the predictive performance was much worse (same as simple MLE). They explain that this model over-smoothes (since it integrates over many topics) and more importantly the discovered topics are divided distinctly between some topics that generate the image segments and some other topics that generate the tags and therefore while the fit is good, the model cannot generate the data properly. The best option was the CorLDA shown in Figure 2-11c in which the image segments are generating the tags by connecting through a uniform distribution and the topic assignments z should generate both the words and the tag. This is the main lesson that we will be following in our design: biasing topics should be done at the individual tokens and by what it is generating and not its prior.

2.2.5.4 Named entities (Entity Topic Models)

Newman et al (2006) propose several approaches to model the named entities along with the text of document. Named entities are generated automatically in a set of New York Times articles. They use the learned conditional probability distributions to do the predictions and evaluate by the average best rank (matching entities to annotated in the best case of multiple runs).

Once again, we focus on the choices made for designing the graphical model: Figure 2-12a is a model similar to GM-LDA that we discussed in the previous section and for the same reasons would create disconnected topics. Also, the topics for the words w and named entities $\widetilde{w}$ do not align properly. Figure 2-12b addresses some of these issues by connecting the two models using x , a Bernoulli variable which determines whether we should use the word topic distribution or the entity topic distribution. Figure 2-12c is the extension of Blei & Jordan (2003) CorLDA for the named entities and in Figure 2-12d the authors allow the connection between models to be through a separate variable x.



a. Conditionally Independent LDA (CI-LDA)

b. Switch-LDA

c. CorrLDA1

d. CorrLDA2

Figure 2-12 Entity Topic Models (Newman et al, 2006)

Except for the differences mentioned between these two types of the information extraction in the section 1.1, this task is related to ours: we can consider the aspects as named entities. Our final model has some similarities with CorrLDA1, however the main issue is that we need to extract the aspects and in all of these models they are assumed to be extracted with a separated tool. We also need to change these models to our semi-supervised setting by dealing with the observed and unobserved cases separately.

2.2.5.5 Pro and Con Phrases in User Reviews

Branavan et al (2008) extend the LDA to incorporate the information from pro/con phrases that accompany some user reviews. Figure 2-13 shows the model which has two segments: the bottom portion is the basic LDA and the top portion of the figure handles the key phrase clustering and assignment to the topics. The combination of the two models is similar to one of the model for named entities: c is a Bernoulli variable that chooses between the two models and decides which one will be generating z. If the topic is generated by η, then it corresponds to the phrase clusters which are the entity aspects. This is essentially biasing the topic distributions in ϕ to correspond to aspects. They perform experiments on a set of online restaurant and cell phone reviews and evaluate by the prediction performance against gold annotations.

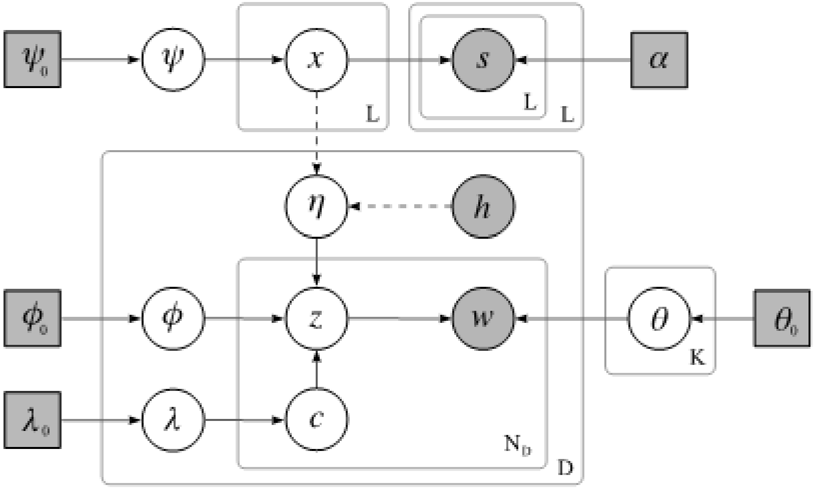


Figure 2-13 Modeling Pros/Cons Phrases in User Reviews (Branavan et al, 2008)

We have the pros and cons in our dataset and thesis original plan was to replicate and compare with the performance of this system. Unfortunately, replicating this system turned out to be quite difficult partly due some implementation details. The authors of this paper provided their data and code toward the end of this thesis and we had to postpone it to future work.

2.2.5.6 Aspect Sentiments

Titov & McDonald (2008) used the model they previous introduced called multi-grain LDA (MG-LDA) in which the model defines two sets of topics: global and local (Figure 2-14a). In this paper, they extend this model to Multi-aspect sentiment model (MAS) that incorporates the supervision available from aspect rating which is explicitly provided in some websites (e.g., hotel reviews from tripadvisor.com have aspect ratings for “cleanliness”). This rating has been added to the model as observed variable y in Figure 2-14b which is affecting r (selection variable between global and local), z (final topic assignment) and w (observed words).

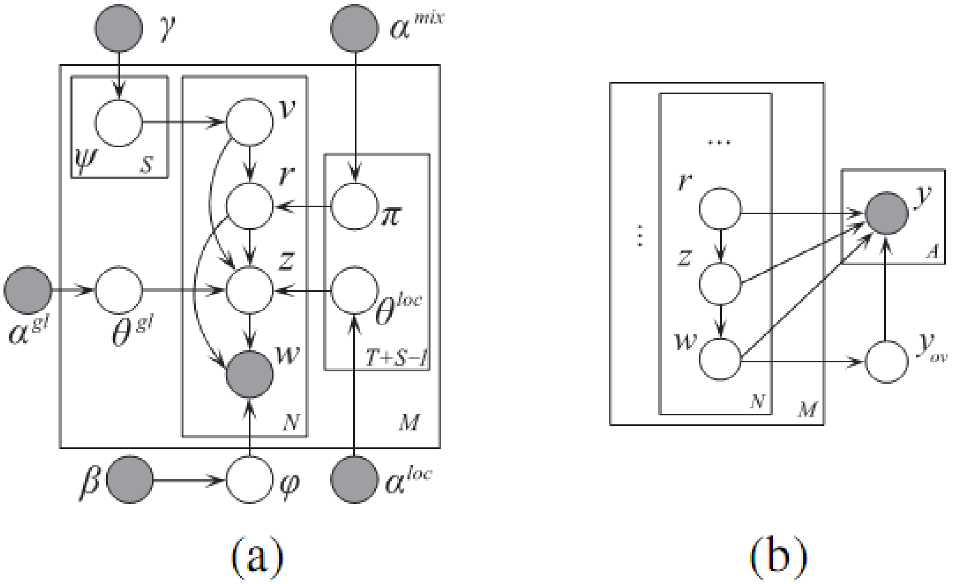


Figure 2-14 Multi-Aspect Sentiment Model (Titov & McDonald, 2008)

2.2.5.7 Document Labels: Supervised LDA (SLDA)

Blei & McAuliffe (2007) introduced the supervised LDA which is doing a regression task on a set of numerically labeled document. Figure 2-15 shows the graphical model. They derive a variational method for this model which is very similar to the LDA. Experiments were done with the movie review dataset with sentiment rating for each document and they show performance improvement over LDA measured by the predictive R2.

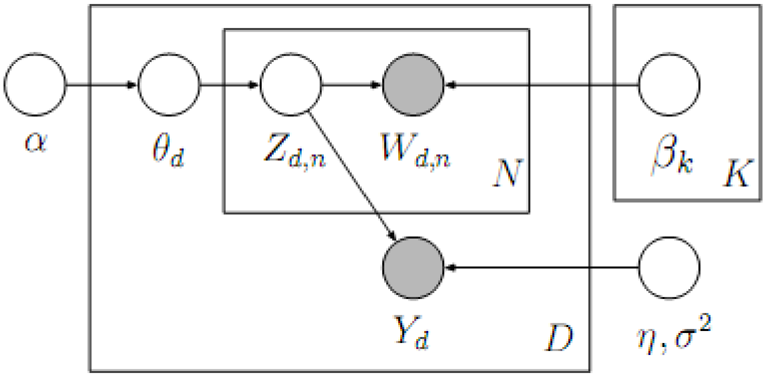


Figure 2-15 Supervised LDA (SLDA) (Blei & McAuliffe, 2007)

In this paper, they further explain why conditioning the response variable Y on Z instead of θ is better: this allows the response to be generated after the document is fully generated and Y is based on the actual counts in Z rather than a mean of the distribution.

Our model is most similar to SLDA with two main differences:

1. Our response variables are at the level of individual words (each span has a label ID and all words in that span will get the same ID).
2. Our response variables are partially observed.

2.2.6 Others Extensions of LDA

Here we discuss other extensions of LDA that we did not use in this thesis but they are relevant to future work in this area.

2.2.6.1 Non-parametric Prior

One critical parameter to pick for LDA is the number of topics (K) which defines the total number of model parameters. A different approach is to let the number of parameters be determined from the data and let the model determine that on its own. This approach is part of a family of methods called ***non-parametric*** Bayesian approaches. To apply this method in LDA, we can use a Dirichlet Process (DP) prior instead of a single Dirichlet distribution on the topic mixture (θ). The approach is described in (Teh et al, 2006) and has been used in many recent papers e.g., (Goldwater et al, 2006; Haghighi & Klein, 2007).

2.2.6.2 Discriminative Training

Lacoste-Julien et al (2008) proposed a discriminative approach using Gibbs sampling for training LDA. The method is basically learning a projection matrix to transform the topic assignments to the label assignments. This is directly related to our work as we are interested in this mapping and we do this through topic labeling using heuristics in the next chapter. Furthermore discriminative approaches may be more suited for segmentation type problems such as ours.

Chapter 3

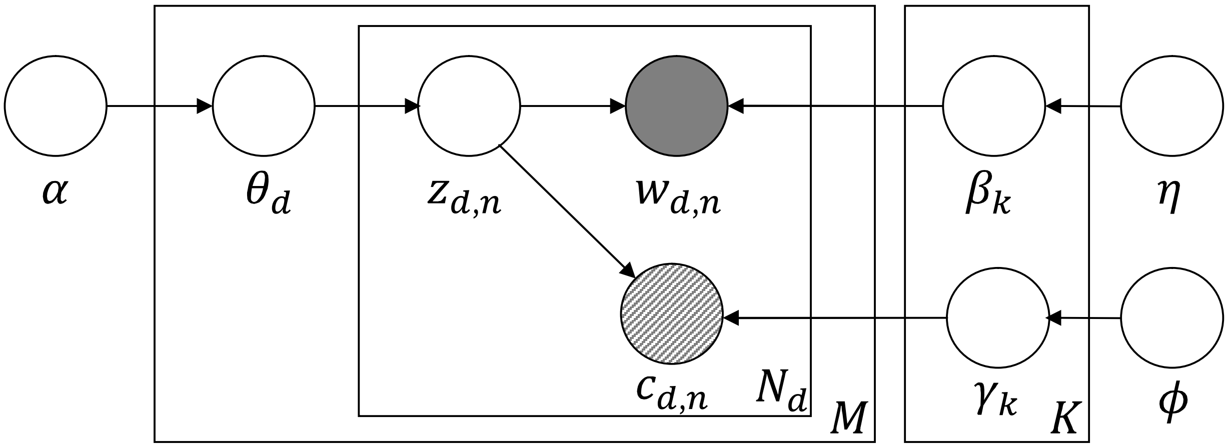
Extracting Entity Aspects

In this chapter, we present our approach to the problem that we defined in chapter 1. We will design this approach based on what we discussed while reviewing related work in chapter 2. We define an extension of LDA to incorporate the side information (from the labeled instances) in our model. The basic idea is simple: LDA is an unsupervised method that will discover the word distributions of a number of topics in the document collection. Ideally, we want these topics to correspond to our goal topics i.e., the entity aspects and therefore our model essentially tries to bias the "natural" topics toward the goal topics using a very small number of labeled instances.

3.1 Semi-Supervised LDA

3.1.1 Definition

We follow the notation in Blei et al (2003). We have a collection of M documents: $\mathcal{D} = \{\boldsymbol{w}_1, \boldsymbol{w}_2, \dots, \boldsymbol{w}_M\}$ and each document contains N_d words: $\boldsymbol{w}_d = (w_{d,1}, w_{d,2}, \dots, w_{d,N_d})$ where $w_{d,n}$ is the word in n-th position of d-th document $\boldsymbol{w}_d$. Each $w_{d,n}$ is an index of a word in our vocabulary which contains V words. Topics are indexed with $k = \{1, 2, \dots, K\}$. For some documents, we have annotations for the aspect which we designate as class labels $c : \mathcal{C}_d = \{c_{d,1}, c_{d,2}, \dots, c_{d,N_d}\}$. Following the discussions related to SLDA in chapter 2, we would like the class labels to be inferred at the word topic assignment to avoid separation of unsupervised and supervised topic distributions. Figure 3-1 show the graphical model in plate notation and the generative process. We have used the hashed fill pattern as “half-shaded” to denote that random variable c has missing values (i.e., some instances have labels) and hence we are in a semi-supervised setting.



Hyper parameters

- α : Topic mixture prior $_{\text{1 x K (Number of topics)}}$
- η : Topic word distribution prior (Smoothing) $_{\text{1 x V (Vocabulary size)}}$
- ϕ : Aspect distribution prior $_{\text{1 x C (Number of labels)}}$

Generative Process

1. For each topic: $k = 1, 2, \dots, K$
   $\beta_k \sim Dirichlet(\eta)$: Choose a topic word distribution $_{\text{1 x V}}$
   $\boldsymbol{\gamma_k} \sim \boldsymbol{Dirichlet(\phi)}$: Choose a class distribution $_{\text{1 x C}}$
2. For each document: $d = 1, \dots, M$
   a. $\theta_d \sim Dirichlet(\alpha)$: Choose a topic mixture $_{\text{1 x K}}$
   b. For each word: $n = 1, \dots, N_d$
      iii. $k = z_{d,n} \sim Multinomial(\theta_d)$: Choose a topic $_{\text{1 x 1}}$
      iv. $w_{d,n} \sim Multinomial(\beta_k)$: Choose a word $_{\text{1 x 1}}$
      v. $\boldsymbol{c_{d,n}} \sim \boldsymbol{Multinomial(\gamma_k)}$: Choose a class $_{\text{1 x 1}}$

Figure 3-1: Semi-Supervised LDA (SS-LDA)

3.1.2 Inference

The likelihood of our document collection based on the factorization shown by the graphical model is the following:

$$p(\mathcal{D}|\alpha,\eta,\gamma) = \prod_{d=1}^{M} p(\boldsymbol{w}_d|\alpha,\eta,\phi) = \prod_{d=1}^{M} \int p(\theta_d|\alpha)p(\beta|\eta)\, p(\gamma|\phi)\, p(\boldsymbol{w}_d|\theta_d,\beta,\gamma) d\phi\, d\eta\, d\theta_d$$

$$p(\boldsymbol{w}_d|\theta,\beta,\gamma) = \sum_{\boldsymbol{z}_d,\boldsymbol{c}_d} p(\boldsymbol{w}_d,\boldsymbol{z}_d,\boldsymbol{c}_d|\theta_d,\beta,\gamma) = \prod_{n=1}^{N_d} \sum_{z_{dn},c_{dn}} p(z_{dn}|\theta_d)p(c_{dn}|\gamma,z_{dn})p(w_{dn}|\beta,z_{dn})$$

Similar to basic LDA, the coupling between θs and βs is causing the likelihood integral be intractable (Blei et al, 2003). We have implemented and tested both Variational method (Blei et al, 2003) and Gibbs Sampling (Griffiths & Steyvers, 2004) and decided to use the Gibbs Sampling for all experiments because the results were better and performance was almost the same (see section 2.2.3 for discussion about the various approximate methods).

For Gibbs Sampling we need to sample each latent variable. The following are the Gibbs updates for each sampling rounds. We use "…" to denote "all other variables":

$$p(\theta_d | \ldots) \propto p(\theta_d|\alpha) p(\boldsymbol{z}_d|\theta_d) = p(\theta_d|\alpha) \prod_{n=1}^{N_d} p\left(z_{d,n}\middle|\theta_d\right) \sim Dir\left(\alpha + count\left(z_{d,n} = k\right)\right)$$

This is due to conjugacy of multinomial $p\left(z_{d,n}\middle|\theta_d\right)$ to Dirichlet $p(\theta_d|\alpha)$. Each of the K parameters of the posterior Dirichlet distribution is updated based on the observed count of topic assignments in document (zs). This posterior needs to be then sampled for each document.

$$p(\beta_k | \ldots) \propto p(\beta_k|\eta) p(\boldsymbol{w}_d|\beta, \boldsymbol{z}_d)$$

$$\propto p(\beta_k|\eta) \prod_{n=1}^{N_d} p(w_{d,n}|\beta, z_{d,n}) \sim Dir\left(\eta + count\left(w_{d,n} = v, z_{d,n} = k\right)\right)$$

In this case, the posterior Dirichlet has V parameters, one for each word in the vocabulary. After the update, it is sampled for each topic.

$$p(z_{d.n} | \ldots) \propto p(z_{d.n}|\theta_d) p(w_{d,n}|\beta, z_{d,n}) p(c_{d,n}|\gamma, z_{d,n})$$

Each distribution on the right hand side is a multinomial and all parameters are known. To find the posterior, for each document, we need to set each possible assignment of z (1 through K) and then normalize. Then sample the posterior for a new $z_{d,n}$ for each word in the document. This is the most time consuming update as it needs sampling of each word in the corpus and each possible assignment of the topics. The time complexity is $O(K. \sum_{d=1}^{M} N_d)$.

For the classes, we have a similar Dirichlet update:

$$p(\gamma_k| \dots) \propto p(\gamma_k|\phi)p(\boldsymbol{c}_d|\gamma_k)$$

$$= p(\gamma_k|\phi)\prod_{n=1}^{N_d} p(c_{d,n}|\gamma_k, z_{d,n}) \sim Dir\left(\phi + count(c_{d,n} = c, z_{d,n} = k)\right)$$

Lastly, the class $c_{d,n}$ is either observed which we will ignored for the sampling or sampled from the multinomial:

$$p(c_{d,n}| \dots) \propto p(c_{d,n}|\gamma, z_{d,n}) \sim Mult(\gamma_{z_{d,n}})$$

Sampling each variable individually is time consuming and can lead to some instability. Griffiths & Steyvers (2004) suggested a faster procedure called Collapsed Gibbs Sampling for the basic LDA which shows that it is sufficient to sample each z given all other zs and other variables. This is achieved by integrating out the θs and βs and then using the final sample to estimate them separately. Here we derive a similar procedure for our SS-LDA with some minor modifications ($\boldsymbol{z}_{-i}$ is all assignment of $\boldsymbol{z}$ except z_i). In these equations j and k used to index topics and i to index the words or latent variables z:

$$p(z_i|\boldsymbol{z}_{-i}, \boldsymbol{w}, \boldsymbol{c}) \propto \; p(w_i|\boldsymbol{z}, \boldsymbol{c}, \boldsymbol{w}_{-i})p(c_i|\boldsymbol{z}, \boldsymbol{w}, \boldsymbol{c}_{-i})p(z_i|\boldsymbol{z}_{-i})$$

$$p(w_i|\boldsymbol{z}, \boldsymbol{c}, \boldsymbol{w}_{-i}) = p(w_i|\boldsymbol{z}, \boldsymbol{w}_{-i}) = \int p(w_i|z_i, \beta)p(\beta|\boldsymbol{w}_{-i}, \boldsymbol{z}_{-i})d\beta = \frac{n_{-i,j}^{w_i} + \eta}{n_{-i,j} + V\eta}$$

$$p(c_i|\boldsymbol{z}, \boldsymbol{w}, \boldsymbol{c}_{-i}) = p(c_i|\boldsymbol{z}, \boldsymbol{c}_{-i}) = \int p(c_i|z_i, \gamma)p(\gamma|\boldsymbol{c}_{-i}, \boldsymbol{z}_{-i})d\gamma = \frac{n_{-i,j}^{c_i} + \phi}{n_{-i,j} + C\phi}$$

$$p(z_i|\boldsymbol{z}_{-i}) = \int p(z_i|\theta)p(\theta|\boldsymbol{z}_{-i})d\theta = \frac{n_{-i,j}^{d_i} + \alpha}{n_{-i}^{d_i} + K\alpha}$$

Here is what each of the count functions mean. In all cases, the current word is not considered in the count (hence the index $-i$)

- $n_{-i,j}^{w_i}$ is the number of times word w_i is assigned to topic *j*
- $n_{-i,j}$ is the total number of words assigned to topic *j*
- $n_{-i,j}^{c_i}$ is the number of times class c_i is used with topic *j*
- $n_{-i,j}^{d_i}$ is the number of words in document d_i assigned to topic
- $n_{-i}^{d_i}$ is the number of words in document d_i (its length)

The final collapsed sampling equation is:

$$p(z_i|\boldsymbol{z}_{-i}, \boldsymbol{w}, \boldsymbol{c}) = \frac{n_{-i,j}^{w_i} + \eta}{n_{-i,j} + V\eta} \frac{n_{-i,j}^{c_i} + \phi}{n_{-i,j} + C\phi} \frac{n_{-i,j}^{d_i} + \alpha}{n_{-i}^{d_i} + K\alpha}$$

After each round, the other latent parameters can be estimated using the following equations using a vector notation:

$$\widehat{\boldsymbol{\beta}}_k = \frac{\left(n_k^{w_0}, n_k^{w_1}, \ldots, n_k^{w_V} \right) + \eta}{n_k + V\eta}$$

$$\widehat{\boldsymbol{\gamma}}_k = \frac{\left(n_k^{c_0}, n_k^{c_1}, \ldots, n_k^{c_C} \right) + \phi}{n_k + C\phi}$$

$$\widehat{\boldsymbol{\theta}}_d = \frac{(n_0^d, n_1^d, \ldots, n_K^d) + \alpha}{n^d + K\alpha}$$

In case of observed labels, we use their values in these equations, otherwise we sample them first from the appropriate multinomial (as shown in the individual sampling equations) and then use the sampled value in the collapsed fashion. This is how we deal with *c* being partially observed (or equivalently partially latent).

Gibbs sampling will start by assigning random topic for all latent variables $\boldsymbol{z}$ and random labels to the unobserved portion of $\boldsymbol{c}$ and then keep iterating until the moving average of perplexity seem to become stable. Section 3.3 will provide more experimental details for Gibbs Sampling.

3.2 Evaluation

Evaluation is always a difficult task in topic models. Here define the typical evaluation metric (perplexity) as well as other heuristics we use to provide prediction and ensure the model is performing as is expected.

3.2.1 Perplexity

This measure is typically used for evaluation latent variable models and it is an indicator of how well the model fits the unseen data after its parameters are learned. Given a test set $\mathcal{D}_{test} = \{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_J\}$, perplexity is the inverse of geometric average of per word likelihood:

$$perplexity(\mathcal{D}_{test}) = \frac{\sum_{d=1}^{J} \log\, p(\boldsymbol{w}_d)}{\sum_{d=1}^{J} N_d}$$

Calculating the document likelihood $p(\boldsymbol{w}_d)$ requires the same intractable inference that we originally faced (normalizing factor needs to find all possible assignment of topics to words). We follow (Griffiths & Steyvers, 2004) to approximate $p(\boldsymbol{w}_d)$ by $p(\boldsymbol{w}_d|\boldsymbol{z}_d)$ where $\boldsymbol{z}_d$ is a posterior sample from our model. We average[3] this over a number of different samples for more stable result (since any single sample may not be likely). Converting to our notation (Figure 3-1):

$$p(\boldsymbol{w}|\boldsymbol{z}) = \left(\frac{\Gamma(V\eta)}{\Gamma(\eta)^V}\right)^K \prod_{k=1}^{K} \frac{\prod_v \Gamma(n_k^v + \eta)}{\Gamma(n_k + V\eta)}$$

[3] This is typically a harmonic mean but we found the same results from arithmetic means as well which means the variance is low.

n_k^v is the number of time word v is assigned to topic k and total number of words assigned to topic k or $n_k = \sum_v n_k^v$. The problem with this measure is the typical problem with semi-suprvise learning: it is possible that there is little or no correlation between minimizing the perplexity (or maximizing likelihood) and minimizing the error rate for a specific task. As a result, we need to provide other evaluations of our model which fortunately is possible because of the presence of labeled data.

For each methods, we ran 5 runs of Gibbs sampling for various number of topics and averaged the result for train set (random 2/3 of data picked for each run) and test set (remaining 1/3 of data). In this case, SS-LDA is given all 50 labeled instances.

3.2.2 Labeling Topics

In section 3.1.2, we showed how this assignment can be used to estimate the model parameters, but we eventually interested in predicting the labels ($\boldsymbol{c}$) from this topic assignments ($\boldsymbol{z}$). We have K topics and C span labels and are interested in obtaining a mapping between them. In this model, we can simply use the labels ($\boldsymbol{c}$) from the model after fitting but these labels are not available in the basic LDA and we also show that by transforming topic distributions , we can obtain better prediction results.

3.2.3 Using label-topic counts

Using the manual gold annotations, we can estimate $P(c|z)$ and then label each topic with the label that maximizes this likelihood:

$$\forall t \in \{1,2,\dots,T\}, \qquad l_t = \mathrm{argmax_c}\ P(c|z) = \mathrm{argmax_c} \frac{n(c \wedge z)}{n(z)}$$

Once the topics are labeled, we can find the label of each point based its topic labels as can be obtained from the Gibbs sample $(\boldsymbol{z})$. We measure how well these predicted labels match the labels from our test set using precision (P) and recall (R). Result reported as F1-measure which is the harmonic mean of the precision and recall: $F_1 = \frac{2PR}{P+R}$

Evaluating for each word may be considered a harsh evaluation for the model although it is a realistic one because it is expected to make the prediction for each span and this allows some partial credit. It is possible to consider less severe penalty by “soft” assigning the labels to topics or count the number of segments matched instead of each work.

3.2.4 Using the word distribution distances

Another way to evaluate is to use the topic word distributions to label each topic. We create an estimate of the true topic distributions $P^*(w|c)$ and then assign labels to each topic based on distance to this distribution:

$$\forall t \in \{1,2,\dots,T\}, \qquad l_t = \mathrm{argmax_c}\ D(P^*(w|c), P(w|t))$$

$D(P_1, P_2)$ is a distance function between two probability distributions for which there are many options available. We have experimented with many possibilities, namely: Kullback-Leibler (KL) divergence, , Cross Entropy, Jaccard distance, Hamming distance, Sum of absolute differences (L1), Sum of squared differences (L2) and Normalized dot product (Cosine) but reported the results only for only two of them.

1. **Jensen–Shannon (JS) divergence** (Lin, 1991)

$$JS(P_1, P_2) = \frac{KL(P_1||Q) + KL(P_2||Q)}{2} = \frac{1}{2}\sum_x \left(P_1(x)\log\frac{P_1(x)}{Q(x)} + P_2(x)\log\frac{P_2(x)}{Q(x)}\right)$$

Where $Q(x) = \frac{1}{2}\big(P_1(x) + P_2(x)\big)$ for $\forall x$.

2. **Euclidean distance (L2)**

$$L_2(P_1, P_2) = \sqrt{\sum_x \left(P_1(x) - P_2(x)\right)^2}$$

Of course, the true topic distribution $P^*(w|c)$ is not known, therefore we need to estimated. Since we only have a few labeled examples, an MLE estimate (i.e., using unigram counts) of parameters of this multinomial probably distribution over a large vocabulary is very biased. Even assuming that we can estimate the true distribution with sufficient amount of data, there is still the concern that multinomial distribution is making independence assumptions between the outcomes which is clearly not correct in the language. In other words, we implicitly make assumption that Naïve Bayes model is the target model for true word distributions which is clearly not the optimal evaluation target.

As an alternative, there has been some work on estimating the probability distribution by direct human annotations e.g., (Mann & McCallum, 2008) which can be explored here to improve this evaluation method.

3.3 Experimental Setup

3.3.1 Dataset

We considered using the dataset from Hu & Liu (2004) which consists of 307 product reviews (1711 sentences) from Amazon.com for 2 digital cameras, a cell phone, an MP3 player and a DVD player. They are manually annotated by the product attributes (called features in their paper) and the polarity of the opinion toward them. Below is an example:

> speakerphone[+3],radio[+3]##the speakerphone , the radio , all features work perfectly .

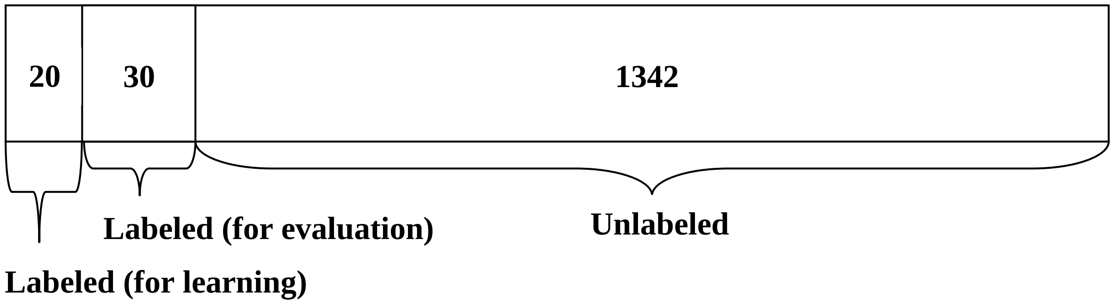


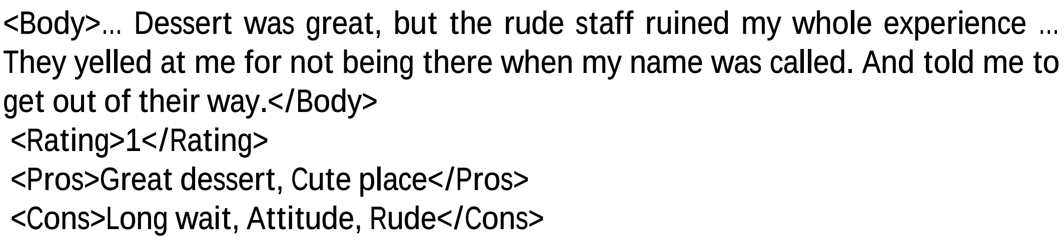
Figure 3-2 Dataset

We learned that we cannot use this dataset for several reasons:

1. We are interested to extract parts of documents that refer to the aspects. While this dataset had the aspect assignments, the span for each aspect was not marked. In the domain of product reviews, the aspect is often explicitly mentioned (as in the example above) but as we showed in section 1.2, this may not be the case in other domains. This data can be used when our model is used at the document level (such as in Blei & McAuliffe (2007)).
2. Labels are redundant (e.g., “picture” and “picture quality”) and therefore the assignments are seriously sparse: there are a total of almost 458 labels are assigned in the whole dataset, 277 only to one sentence and 391 to 5 sentences or less.

We decided to start annotating a set of online reviews. The original data was collected from CitySearch.com with 50,000 users review for New York City restaurants. Reviews accompanied have pros and cons phrases and overall rating.

Below is an example:



```
<Body>… Dessert was great, but the rude staff ruined my whole experience …
They yelled at me for not being there when my name was called. And told me to
get out of their way.</Body>
 <Rating>1</Rating>
 <Pros>Great dessert, Cute place</Pros>
 <Cons>Long wait, Attitude, Rude</Cons>
```

To make the dataset size manageable for the computationally intensive algorithms of our model, we picked 1392 reviews from a random set of restaurants. The dataset is not balanced for the ratings because we do not use them in our experiments. Furthermore, the pros and cons can be helpful in this task as shown in Branavan et al (2008) but we do not use them here.

We needed small amount of manual annotations for our method and also evaluation. 50 of the reviews were manually annotated, 20 of them with two annotators and Kappa (Cohen, 1960) was 0.707 (See Appendix A for more details).

To build our vocabulary, we have used space tokenization and eliminated all punctuations, stop words or words occurring in less than 5 documents. This will leave us with 1520 words in our vocabulary.

Figure 3-2 shows how the dataset was used. We always use the same 20 instances for learning (in the training of the model and also for labeling the topics) and the same 30 instances for testing. Unfortunately, cross validation is not possible for our case as the number of labeled instances is always small.

3.3.2 Gibbs Sampling

As it was discussed, convergence in Gibbs sampling cannot be observed. Many heuristics used in the literature. We monitored the test set perplexity and stopped when the change was less than a small threshold and observed than in most cases the Gibbs sampling converges after 500 iterations. For all experiments, we fixed the number of iterations at 500.

Another issue is the fact that any single sample from posterior distribution may be very unlikely and since we are estimating the parameters with samples, we have no way of knowing this. It is common that when possible to use an average of samples instead of a single one. For obtaining the perplexity at any iteration, we calculated the average of the last 100 samples. Also we skip over the 100 initial samples[4] (this is called "burn in" phase).

Correlation between samples is another potential problem when doing Gibbs sampling: since each sample is generated based on the previous sample result this can create a bias in the parameters. To avoid this problem, we can skip samples. We have tried this but it did not have any effect in our results so in final result we are using all samples after the burn in period.

[4] The results are not sensitive to the burn-in period size especially if the total number of iterations is large enough (500 in our case).

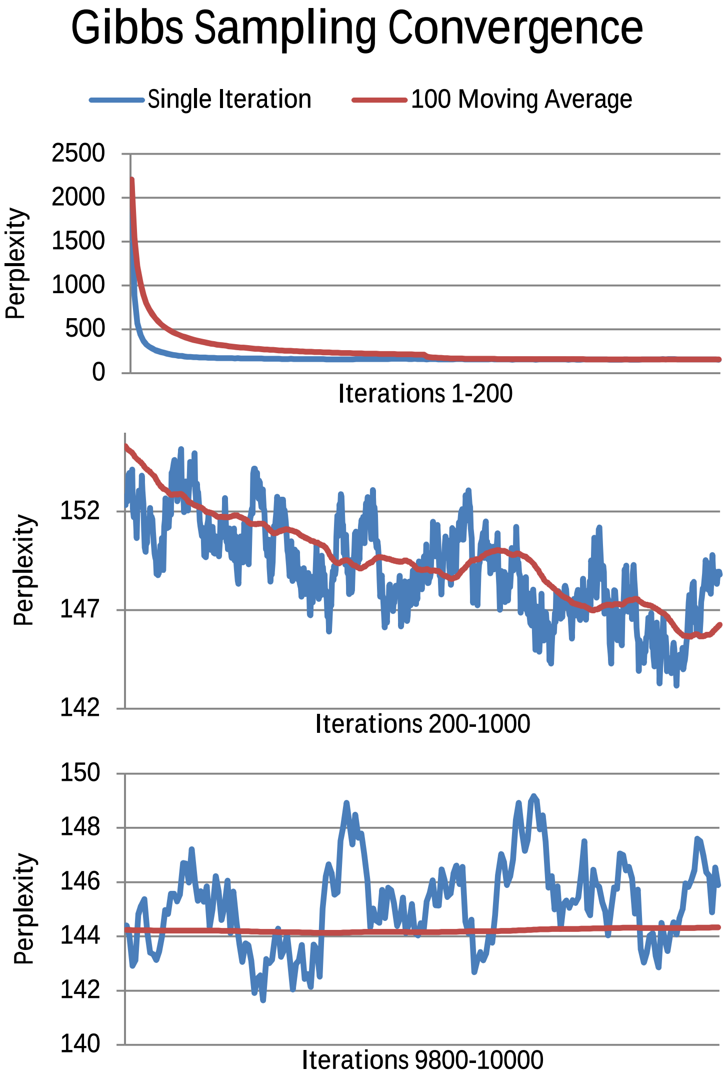


Figure 3-3 Gibbs Sampling Convergence (K=10)

3.4 Results

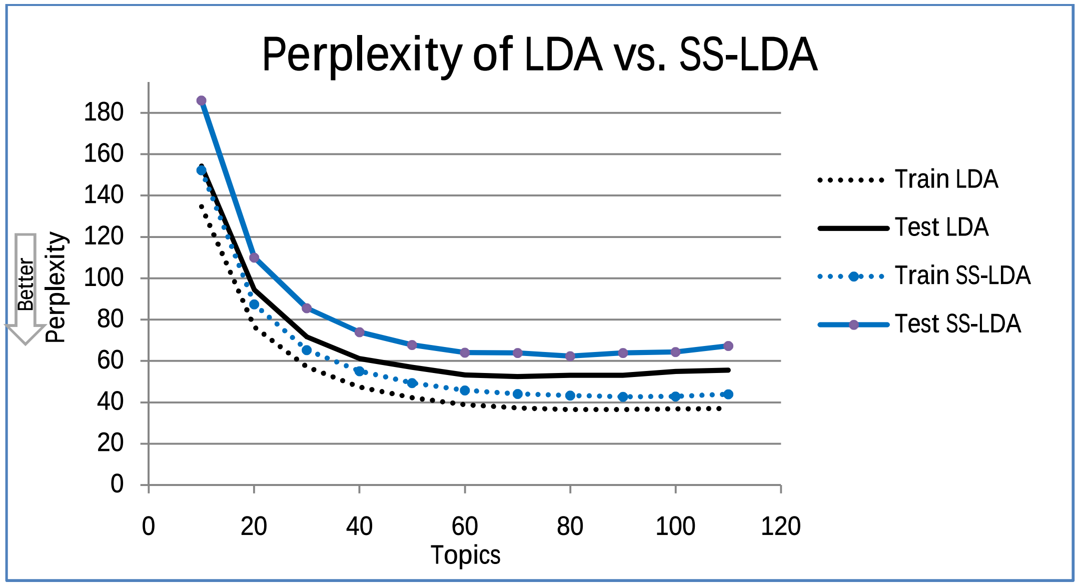


Figure 3-4 Perplexity of LDA vs. SS-LDA

Figure 3-4 shows a comparison of perplexity the LDA vs. SS-LDA for various numbers of topics. Each point is a 10-run average number. Training set is $\frac{2}{3}$ and testing set is $\frac{1}{3}$ of data and each run of Gibbs sampling is run for 500 iterations (running time of each run on Intel Duo Core 2.0 GHz was from 5-30 minutes depending on the number of topics).

Perplexity of the SS-LDA is slightly higher than the equivalent LDA. We believe this is resulted from the bias that we create in the topic distributions to provide a better fit to the labels and expect that this shift provides a better prediction result. The model is fitting more parameters to more complex data (with labels vs. no labels). It can be seen that the perplexity plateaus around 60 topics. We use $K = 60$ for some of the future experiments based on this observation.

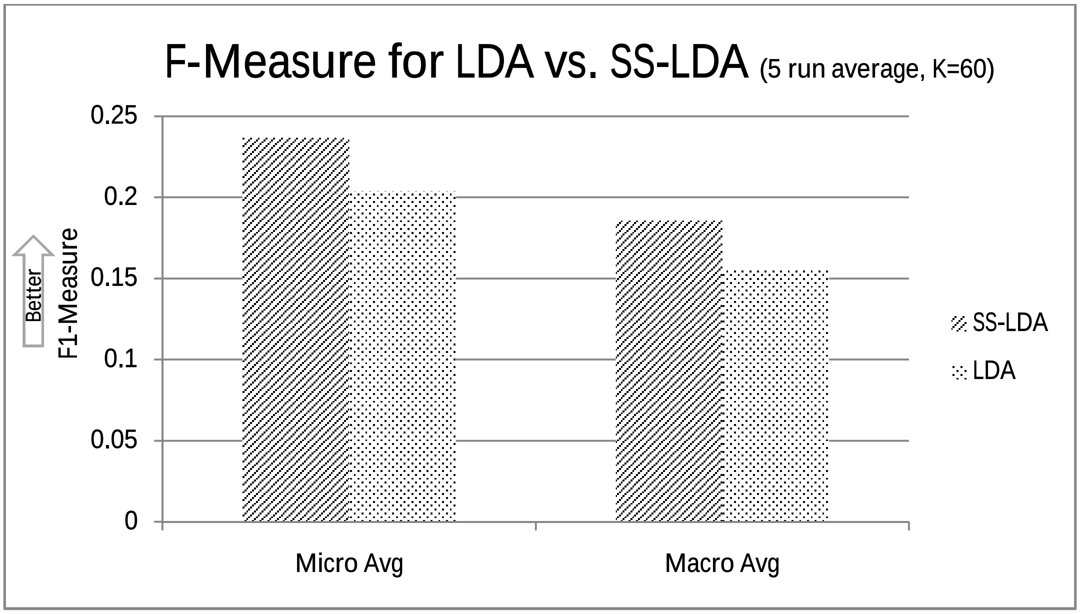


Figure 3-5 F-Measure for LSA and SS-LDA

Figure 3-5 shows the prediction power of LDA vs. SS-LDA. As discussed before, this is measured at the word level and therefore the overall numbers are low in both cases. We show both macro averaged result (i.e., averaging over F1 values for each aspect) and micro averaged result (i.e., calculating the F1 measure by considering all aspect as one class and combine the numbers).

In both methods of calculating the F-measure, we can see that SS-LDA improves its prediction performance in the presence of the small number labeled instances (p-value<0.01 using sign test).

It is important to observe if there is any benefit for this semi-supervised method by changing the amount of unlabelled data. We show this result in Figure 3-6 where in both cases the same number of labeled instances is used for learning and labeling (i.e., 20). It is interesting to note that although LDA can benefit from more unlabelled data but SS-LDA improves its prediction much more than LDA.

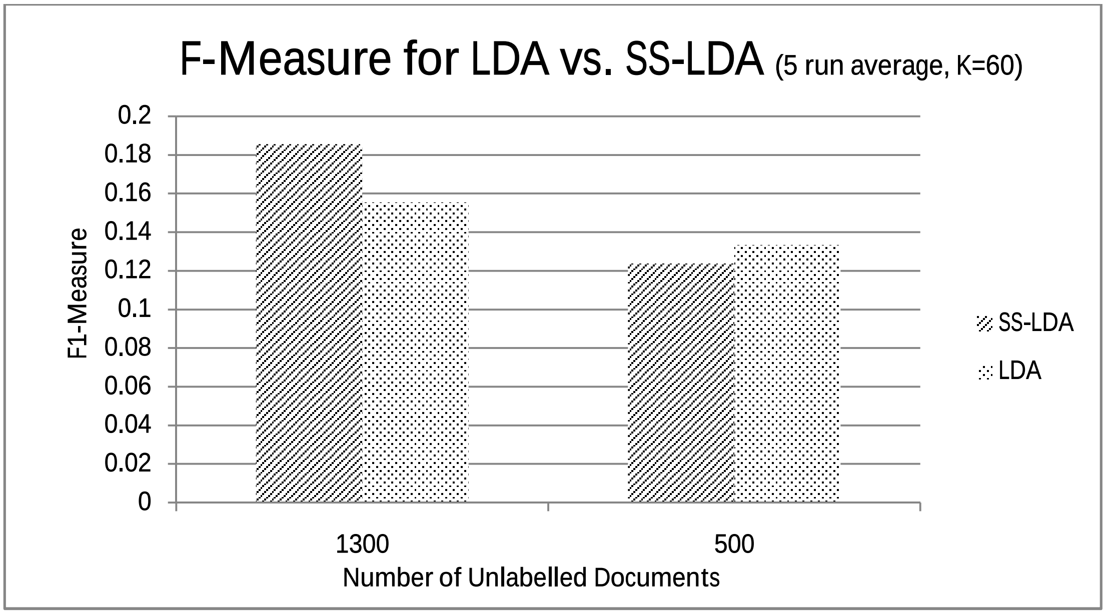


Figure 3-6 Changing the amount of unlabelled data for LSA vs. SS-LDA

Another parameter is the number of topics and we show its effect in Figure 3-7. Surprisingly, increasing the number of topics improves F-measure. We believe that this result is not reliable and needs further investigation: since we are only using 20 labeled instances to label topics and then number of topics is increased beyond 100, we observed that we fail to label some topics based on the labeled instances (no word occurrences). Furthermore, there is a problem with the over fitting which may not become apparent because of small size of our test set with labels. The significance tests for comparison of (LDA 500 < LDA 1300) and (SS-LDA 500 < LDA 500) returns (0.01 < p-value < 0.05, not significant) while (LDA 1300 < SS-LDA 1300) has 0.001 < p-value < 0.01 and finally, (SSLDA500 < SSLDA1300) has 0.0001 < p-value < 0.001 (very significant).

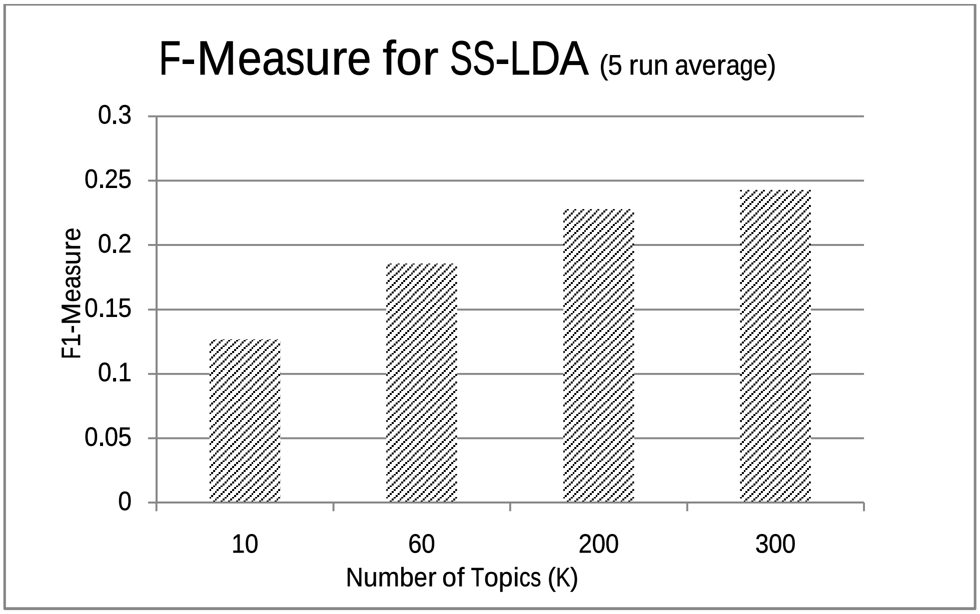


Figure 3-7 Changing the number of topics for SS-LDA

We also compared the two models based on how successful they are in biasing the topic word distributions of LDA (despite the problems with this evaluation that was described in previous section). Figure 3-8 shows the difference between the LDA and SS-LDA when measured through the JS and Figure 3-9 shows same evaluation using L2 metric. In both cases, it can be seen that SS-LDA distributions are closer to the estimated true topic distributions than the ones from LDA.

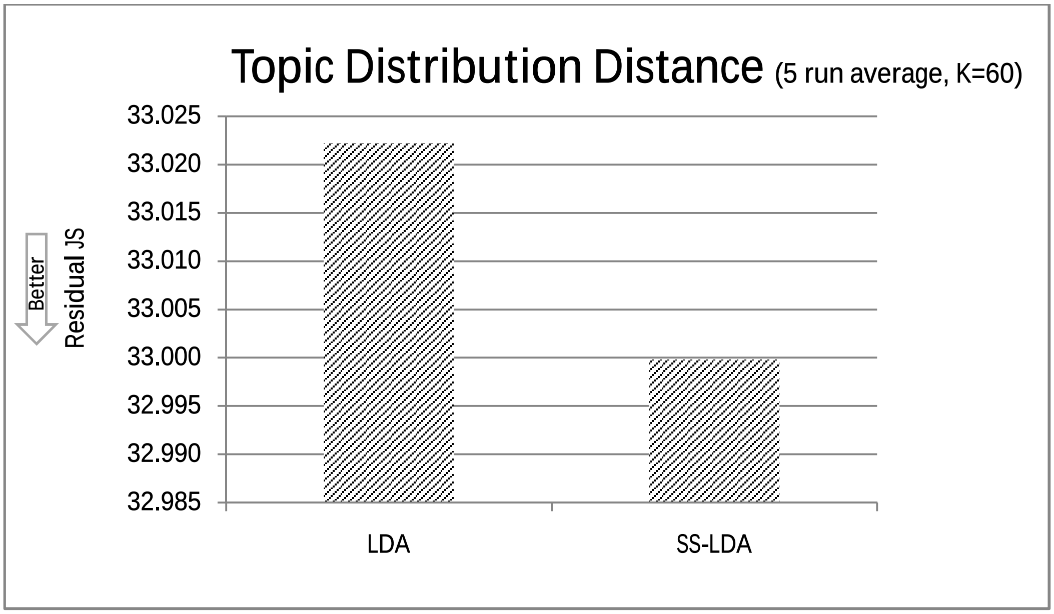


Figure 3-8 Topic Distribution distance change using Jensen Shannon Divergence (JS)

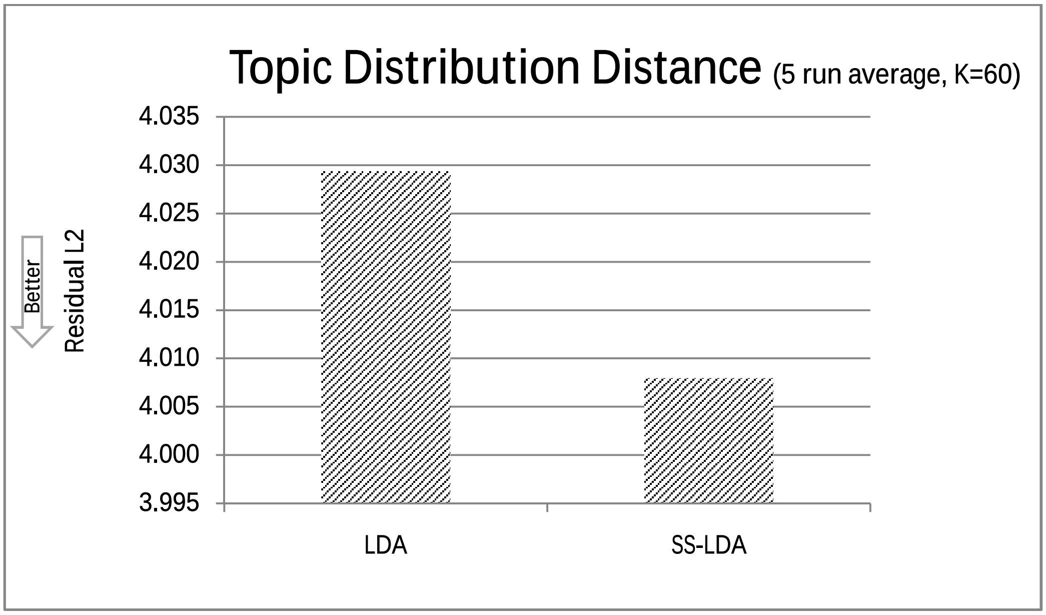
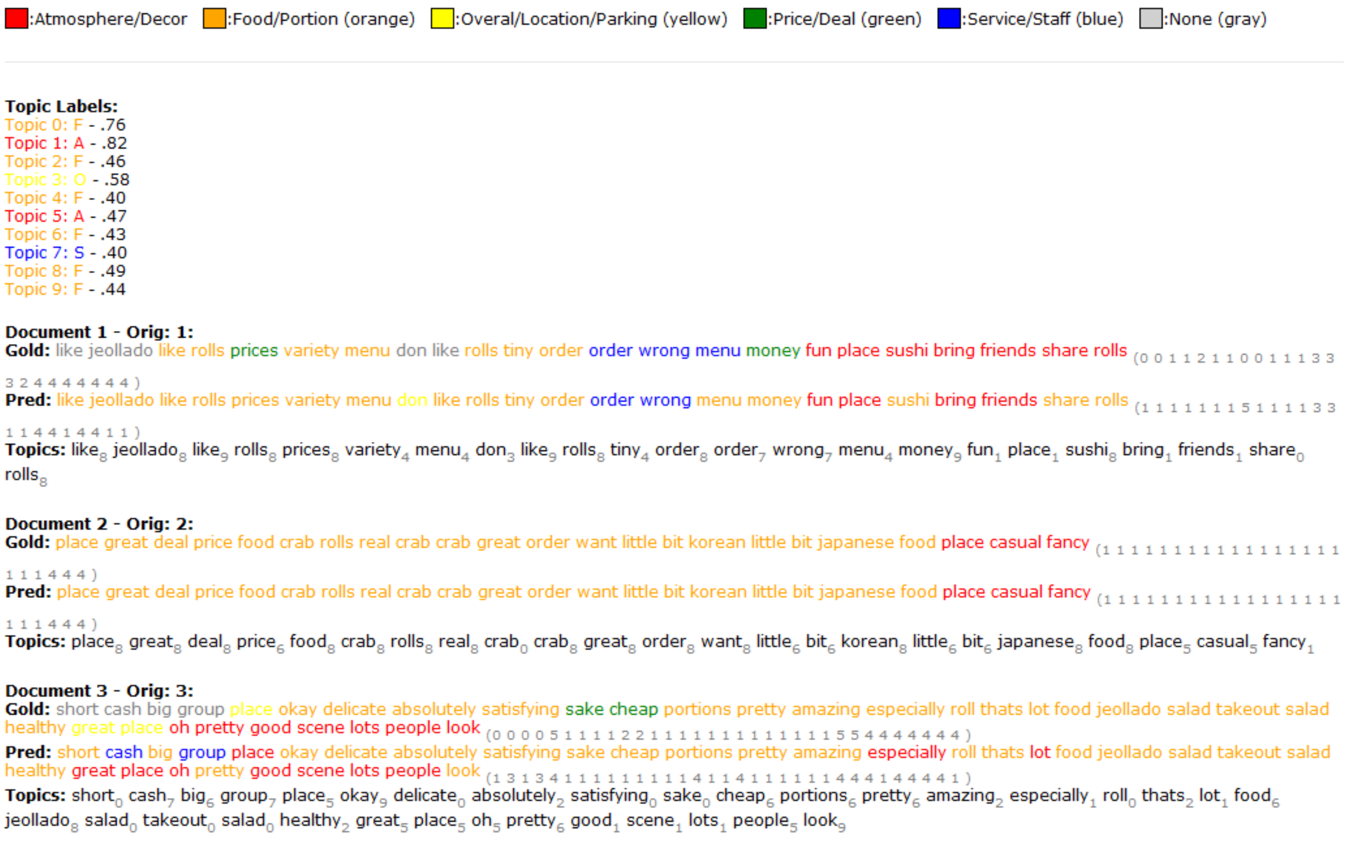


Figure 3-9 Topic Distribution distance change using Euclidean Distance (L2)

3.5 Conclusion

We showed a way of using a small number of labeled instances (i.e., side information) to change the topic distributions that basic LDA can discover from the data.

It has been known that generative methods need fewer labeled instances for convergence than discriminative methods. We did not have the time to empirically compare our results to a discriminative approach but considering very small number of examples (50 out of 1400), we believe they may not perform very well.



🟥:Atmosphere/Decor 🟧:Food/Portion (orange) 🟨:Overal/Location/Parking (yellow) 🟩:Price/Deal (green) 🟦:Service/Staff (blue) ⬜:None (gray)

Topic Labels:
Topic 0: F - .76
Topic 1: A - .82
Topic 2: F - .46
Topic 3: O - .58
Topic 4: F - .40
Topic 5: A - .47
Topic 6: F - .43
Topic 7: S - .40
Topic 8: F - .49
Topic 9: F - .44

Document 1 - Orig: 1:
Gold: like jeollado like rolls prices variety menu don like rolls tiny order order wrong menu money fun place sushi bring friends share rolls (0 0 1 1 2 1 1 0 0 1 1 1 3 3 3 2 4 4 4 4 4 4 4)
Pred: like jeollado like rolls prices variety menu don like rolls tiny order order wrong menu money fun place sushi bring friends share rolls (1 1 1 1 1 1 1 5 1 1 1 1 3 3 1 1 4 4 1 4 4 1 1)
Topics: like$_{8}$ jeollado$_{8}$ like$_{9}$ rolls$_{8}$ prices$_{8}$ variety$_{4}$ menu$_{4}$ don$_{3}$ like$_{9}$ rolls$_{8}$ tiny$_{4}$ order$_{8}$ order$_{7}$ wrong$_{7}$ menu$_{4}$ money$_{9}$ fun$_{1}$ place$_{1}$ sushi$_{8}$ bring$_{1}$ friends$_{1}$ share$_{0}$ rolls$_{8}$

Document 2 - Orig: 2:
Gold: place great deal price food crab rolls real crab crab great order want little bit korean little bit japanese food place casual fancy (1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 4 4 4)
Pred: place great deal price food crab rolls real crab crab great order want little bit korean little bit japanese food place casual fancy (1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 4 4 4)
Topics: place$_{8}$ great$_{8}$ deal$_{8}$ price$_{6}$ food$_{8}$ crab$_{8}$ rolls$_{8}$ real$_{8}$ crab$_{0}$ crab$_{8}$ great$_{8}$ order$_{8}$ want$_{8}$ little$_{6}$ bit$_{6}$ korean$_{8}$ little$_{6}$ bit$_{6}$ japanese$_{8}$ food$_{8}$ place$_{5}$ casual$_{5}$ fancy$_{1}$

Document 3 - Orig: 3:
Gold: short cash big group place okay delicate absolutely satisfying sake cheap portions pretty amazing especially roll thats lot food jeollado salad takeout salad healthy great place oh pretty good scene lots people look (0 0 0 0 5 1 1 1 1 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 5 5 4 4 4 4 4 4 4)
Pred: short cash big group place okay delicate absolutely satisfying sake cheap portions pretty amazing especially roll thats lot food jeollado salad takeout salad healthy great place oh pretty good scene lots people look (1 3 1 3 4 1 1 1 1 1 1 1 1 1 4 1 1 4 1 1 1 1 1 1 1 4 4 4 1 4 4 4 4 1)
Topics: short$_{0}$ cash$_{7}$ big$_{6}$ group$_{7}$ place$_{5}$ okay$_{9}$ delicate$_{0}$ absolutely$_{2}$ satisfying$_{0}$ sake$_{0}$ cheap$_{6}$ portions$_{6}$ pretty$_{6}$ amazing$_{2}$ especially$_{1}$ roll$_{0}$ thats$_{2}$ lot$_{1}$ food$_{6}$ jeollado$_{8}$ salad$_{0}$ takeout$_{0}$ salad$_{0}$ healthy$_{2}$ great$_{5}$ place$_{5}$ oh$_{5}$ pretty$_{6}$ good$_{1}$ scene$_{1}$ lots$_{1}$ people$_{5}$ look$_{9}$

Figure 3-10 Sample HTML Output of the program with topic and label predictions

Chapter 4

Future Work

We showed an approach in biasing the LDA topics toward a goal, represented through a few labeled instances. There are many different approaches that can be attempted for this task and unfortunately, in this thesis, we did not have sufficient time for them. In this section, we outline some ideas in addition to those suggested as future work in the previous chapters.

4.1 Segmentation LDA

One problem with LDA is that there is no coherence in the topics. There is no provision to force the model to designate topic so that the similar topics are preferred to be closer to each other. The exchangeability assumption in LDA makes the model indifferent about the position of the words. We have seen in chapter 2 that some models (such as STM) are trying to change the scope of the exchangeability assumption. We have designed a model to perform the segmentation with the coherence directly modeled. Figure 4-1 shows the graphical model of the proposed approach. The model generates the document segmentation by drawing the proportions ψ_d from a Dirichlet distribution ϕ. Then the length of each segment $l_{d,s}$ is drawn from the multinomial with parameter ψ_d. For each of the S segment, the topic of the segment is fixed (hence the $z_{d,s}$ is the segment plate).

Inference in this model is tricky because the distribution of segment lengths is influencing the topics indirectly and therefore the conditionals cannot be sampled using conventional Gibbs sampling. Experiment with this model requires resolving this issue first.

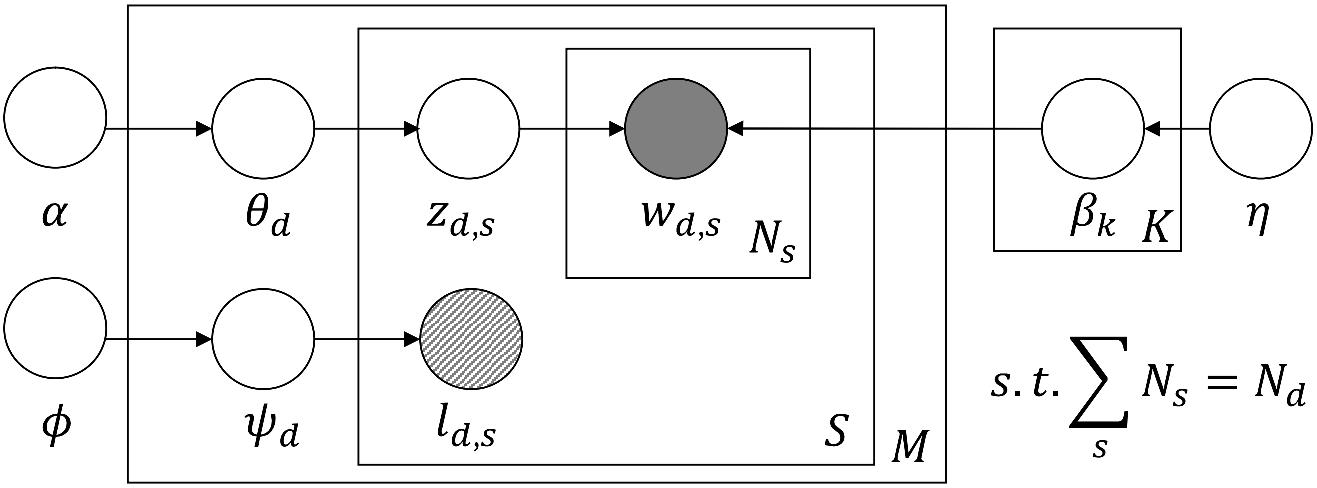


Figure 4-1 Segmentation LDA

4.2 Other Directions

In chapter 2, we mentioned some of the alternate approaches which are the natural next steps for our approach, namely trying different (possibly non-parametric) priors as well as discriminative training as it is more suited to segmentation task.

Our model is ignoring a lot of useful information that can potentially be beneficial toward this task. Polarity and sentiment of the reviews can provide some good clues for discovering the aspects. There may also be some advantage to use some better initial sets of aspect by other types of clustering such as K-Means or Spectral clustering both independently or jointly with SS-LDA.

Traditional n-gram language models or some of the new ones based on HDP (Teh et al, 2006) can also be combined to our model and can lead to some improvement since they are relaxing improving the modeling assumptions.

Bibliography

Blei, D.M. & Jordan, M.I., 2003. Modeling annotated data. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*.

Blei, D. & Lafferty, J., 2006. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*.

Blei, D. & Lafferty, J., 2007. A correlated topic model of Science. In *Annals of Applied Statistics*.

Blei, D.M. & McAuliffe, J., 2007. Supervised topic models. In *Advanced In NIPS*.

Blei, D.M., Ng, A.Y. & Jordan, M.I., 2003. Latent Dirichlet Allocation. In *Journal of Machine Learning Research*.

Boyd-Graber, J. & Blei, D., 2009. Syntactic Topic Models. In *Neural Information Processing Systems*.

Branavan, S., Chen, H., Eisenstein, J. & Barzilay, R., 2008. Learning Document-Level Semantic Properties from Free-text Annotations. In *Proceedings of ACL*.

Chang, J. & Blei, D., 2009. Relational Topic Models for Document Networks. In *Artificial Intelligence and Statistics*.

Cohen, J., 1960. A coefficient of agreement for nominal scales. In *Education and Psychological Measuremen*.

Deerwester, S. et al., 1990. Indexing by latent semantic analysis. In *Journal of the American Society for Information Science*.

Erosheva, E., Fienberg, S. & Lafferty, J., 2004. Mixed-membership models of scientific publications. In *Proc Natl Acad Sci U S A*.

Goldwater, S., Griffiths, T.L. & Johnson, M., 2006. Contextual Dependencies in Unsupervised Word Segmentation. In *Proceedings of Coling/ACL*.

Goldwater, S., Griffiths, T.L. & Johnson, M., 2006. Contextual dependencies in unsupervised word segmentation. In *Proceedings of COLING/ACL*.

Griffiths, T.L. & Steyvers, M., 2004. Finding scientific topics. In *Proc Natl Acad Sci U S A*.

Griffiths, T.L., Steyvers, M., Blei, D.M. & Tenenbaum, J.B., 2005. Integrating topics and syntax. In *Advances in NIPS 17*.

Gruber, A., Rosen-Zvi, M. & Weiss, Y., 2007. Hidden Topic Markov Models. In *Artificial Intelligence and Statistics*.

Haghighi, A. & Klein, D., 2007. Unsupervised Coreference Resolution in a Nonparametric Bayesian Model. In *Association for Computational Linguistics*.

Hofmann, T., 1999. Probabilistic latent semantic analysis. In *Proc. of Uncertainty in Artificial Intelligence, UAI'99*.

Hu, M. & Liu., B., 2004. Mining and summarizing customer reviews. In *Proceedings of SIGKDD*.

Lacoste-Julien, S., Sha, F. & Jordan, M., 2008. DiscLDA: Discriminative learning for dimensionality reduction and classification. In *Proceedings of NIPS 21*.

Levin, E. & Sharifi, M., 2006. Evaluation of Utility of LSA for Word Sense Discrimination. In *Proceedings of HLT/NAACL*.

Lin, J., 1991. Divergence measures based on the Shannon entropy. In *IEEE Transactions on Information Theory*.

Mann, G. & McCallum, A., 2008. Generalized Expectation Criteria for Semi-Supervised Learning of Conditional Random Fields. In *ACL*.

Mccallum, A., Corrada-Emmanuel, Andres & Wang, X., 2005. Topic and Role Discovery in Social Networks. In *Proceeding of IJCAI*.

Minka, T. & Lafferty, J., 2002. Expectation-propagation for the generative aspect model. In *Proceedings of UAI*.

Newman, D., Chemudugunta, C. & Smyth, P., 2006. Statistical entity-topic models. In *Proceedings of the 12th ACM SIGKDD*.

Ng, A.Y. & Jordan, M., 2002. On Discriminative vs. Generative Classifiers: A comparison of logistic regression and Naive Bayes. In *NIPS*.

Nigam, K., McCallum, A., Thrun, S. & Mitchell, T., 2000. Text classification from labeled and unlabeled documents using EM. In *Journal of Machine Learning*.

Popescu, A. & Etzioni, O., 2005. Extracting product features and opinions from reviews. In *Proceedings of HLT/EMNLP*.

Rosen-Zvi, M., Griffiths, T., Steyvers, M. & Smyth, P., 2004. The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*.

Teh, Y.W., Jordan, M.I., Beal, M.J. & Blei, D.M., 2006. Hierarchical Dirichlet Processes. In *Journal of the American Statistical Association*.

Titov, I. & McDonald, R., 2008. A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of the ACL*.

Wang, C., Thiesson, B., Meek, C. & Blei, D., 2009. Markov Topic Models. In *Artificial Intelligence and Statistics*.

Zhu, X., 2005. Semi-supervised learning literature survey. In *Technical Report 1530, ComputerS ciences, University of Wisconsin-Madison*.

Appendix A

Human Annotations

Manual human annotations were needed for evaluation and also testing the performance of the methods in the semi-supervised and supervised settings where the goal (or interest) of the user is expressed in the form of simple annotations of the text.

A.1 Syntax

An annotation for this project is an assignment of a class label to a span of text. Text spans are following conditions:

1. They are arbitrary length and position (i.e., don’t correspond to phrases or sentences)
2. They are non-overlapping
3. Do not necessarily cover the entire document

Annotation is performed on the plain text of the reviews and the file format resembles the files provided by Hu and Liu (2004). Each review is annotated with the following syntax. All sections are option can be elimitated (proceeded to the next by including “|”):

`[CoarseLabel][Sentiment]|[FineLabels]||[Rationales]#[TextSpan]`

- **`CoarseLabel`** can be one of the following five values (single character):

- A (= **A**tmosphere): any reference to décor, ambience, noise, temperature, cleanliness, etc.
- F (= **F**ood): any reference for what was eaten, portion size, menu and the variety, etc.
- P (= **P**rice): any reference to value, “worth it”, etc.
- S (= **S**ervice): any interaction with restaurant employee or their policies (such as wait), etc.
- O (= **O**verall): any overall comment (e.g., whether they recommend it) or reference to other attributes of the restaurant (e.g., location, easy to find, parking place, etc.)
- *Blank* : when the segment is not relevant to the entity or doesn’t express any opinion

- **`Sentiment`** + (for positive), - (for negative), Blank (for not annotated, neutral, mixed or unclear)
- **`FineLabels`** Free form text further refining the coarse and making it more specific. Clarifications given above for coarse labels are examples that can be used.
- **`Rationales`** Words in the text which resulted in making the decision about sentiment (Reference??)

A.2 Examples

[t] Review 1

F+|Deal|great#This place is a great deal for the price and the food they give you.

F+|Authentic|real#Crab rolls are made with real crab, not the imitation crab.

F+||#They also have a great unagi bim bim bap that you must order.

F#Go here if you want a little bit of Korean combined with a little bit of Japanese food.

A|Casual||casual, fancy#Place is casual, not fancy.

[t] Review 2

#Short of cash, with a big group, starving?

O+||#This is the place.

F-|Quality||delicate, delectible#Okay, not the most delicate or delectible, but absolutely satisfying

P+||cheap#and the sake is cheap too.

F#Portions are pretty amazing- especially the Amy roll for $8- damn thats a lot of food for $8. And the Jeollado (seaweed) salad- sometimes I get take-out somewhere else and go get that salad to be healthy and fill me up.

O+||#All in all a great place,

A-||#but not the most refined.

A+|Scene#Oh, also a pretty good scene, lots of people to look at.

[t] Review 4

#Jeollado has seen its good days and bad days...

P#overall the restaurant has great prices for the value.

A|Noise#Loud cafteria seatings and great for large groups.

A|Private Room#There is a private room in the back for parties which include KTV.

[t] Review 36

O#The mini ride at the beginning was a very nice touch.

A#The interior decorating, was out of this world but the only thing I personally did not like was the martians walking around.

F#The food was OK,

O#nothing worth going back for. What I do recommend is going once just to fill your curiosity, you won't regret it.

[t] Review 43

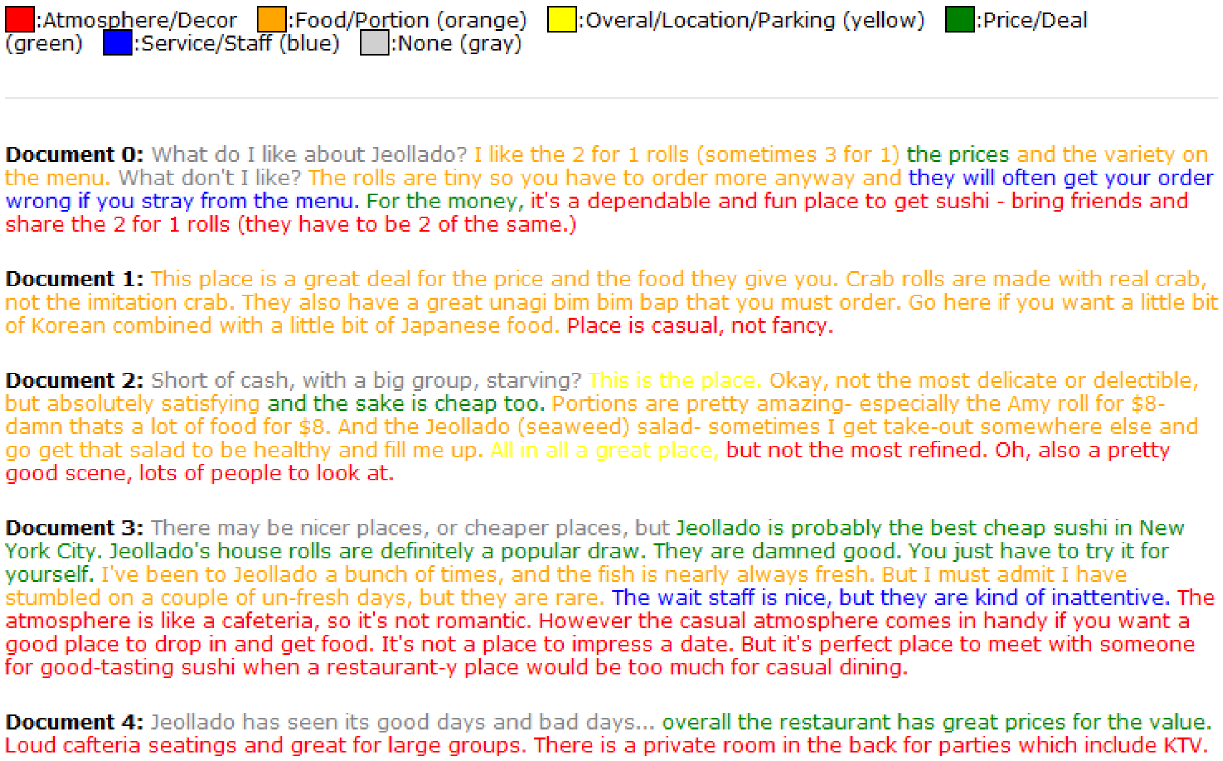
O#Mars 2112 is absolutely miserable.

S#The service is terrible,

F#the food even worse and

P#the prices extremely unworthy.

O#I'm afraid if they turned the lights on everyone including the alien waiters would be scared away...who knows what was living in the carpets.



🟥:Atmosphere/Decor 🟧:Food/Portion (orange) 🟨:Overal/Location/Parking (yellow) 🟩:Price/Deal (green) 🟦:Service/Staff (blue) ⬜:None (gray)

Document 0: What do I like about Jeollado? I like the 2 for 1 rolls (sometimes 3 for 1) the prices and the variety on the menu. What don't I like? The rolls are tiny so you have to order more anyway and they will often get your order wrong if you stray from the menu. For the money, it's a dependable and fun place to get sushi - bring friends and share the 2 for 1 rolls (they have to be 2 of the same.)

Document 1: This place is a great deal for the price and the food they give you. Crab rolls are made with real crab, not the imitation crab. They also have a great unagi bim bim bap that you must order. Go here if you want a little bit of Korean combined with a little bit of Japanese food. Place is casual, not fancy.

Document 2: Short of cash, with a big group, starving? This is the place. Okay, not the most delicate or delectible, but absolutely satisfying and the sake is cheap too. Portions are pretty amazing- especially the Amy roll for $8-damn thats a lot of food for $8. And the Jeollado (seaweed) salad- sometimes I get take-out somewhere else and go get that salad to be healthy and fill me up. All in all a great place, but not the most refined. Oh, also a pretty good scene, lots of people to look at.

Document 3: There may be nicer places, or cheaper places, but Jeollado is probably the best cheap sushi in New York City. Jeollado's house rolls are definitely a popular draw. They are damned good. You just have to try it for yourself. I've been to Jeollado a bunch of times, and the fish is nearly always fresh. But I must admit I have stumbled on a couple of un-fresh days, but they are rare. The wait staff is nice, but they are kind of inattentive. The atmosphere is like a cafeteria, so it's not romantic. However the casual atmosphere comes in handy if you want a good place to drop in and get food. It's not a place to impress a date. But it's perfect place to meet with someone for good-tasting sushi when a restaurant-y place would be too much for casual dining.

Document 4: Jeollado has seen its good days and bad days... overall the restaurant has great prices for the value. Loud cafteria seatings and great for large groups. There is a private room in the back for parties which include KTV.

Figure 4-2 Visualizing the manual annotations

A.3 Process

Two annotators annotated 50[5] reviews, 20 of which were annotated by both. Due to time and resource limitation, only coarse labels were consistently labeled and sometimes the sentiment. While the annotations where for spans of text but we measured the agreement at the level of the document (whether or not certain attribute was assigned to a document or not by an annotator). The table below shows the Kappa measure (Cohen, 1960) for each aspect. Note the average excluding the O (which is a broad category similar to “others”) is 0.79.

[5] We understand this number is small. Actually, more annotations were collected but due to some issues that could not be used. We plan to add more annotations if this research continued. On the other hand, we hope that any semi-supervised method like ours doesn’t need much labeled data as they are expensive to create.

Label	Kappa
A	0.710
F	0.770
O	0.350
P	0.800
S	0.890
Average	0.704

A.4 Open Questions

1. Reviews were selected at random but perhaps more care can be taken to the review length, content language model, or provide balance for various ratings. More complicated sampling method can be applied (as in active learning)
2. Should the rationales be sub-segments of text or an arbitrary set of words?

Appendix B

STAT (Semi-supervised Text Analysis Toolkit)

The program for this thesis is written in Java and then integrated into a new software application which is a machine learning framework. The program was designed as part of the course project for the Software Engineering course with collaborations from Jing Yang, Shilpa Arora and Shima Hideki as the course TAs, Eric Nyberg and Anthony Tomasic as instructors and all other class member's feedback.

For more information about this software package, please visit:

http://seit1.lti.cs.cmu.edu/projects/stat