Cross Collection Aspect Based Opinion Mining Using Topic Models

by
Hemed Hamisi Kaporo

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfilment of
the requirements for the degree of
Master of Science

Sabancı University
July 2018

Cross Collection Aspect Based Opinion Mining Using Topic Models

APPROVED BY:

Prof. Yücel Saygın ....
(Thesis Supervisor)

Asst. Prof. Kamer Kaya ....

Asst. Prof. Ayşe Tosun ....

DATE OF APPROVAL: 31/07/2018

Acknowledgements

This thesis would not be possible without the support of many people in my life. It also cannot be finalized without expressing my gratitude to them.

I would like to express my gratitude and thank my thesis advisor, Prof. Yücel Saygın for his patience and trust. Without his guidance, open-minded discussions and continuous encouragement, this thesis would not be where it is now. Along with Prof. Saygın, an acknowledgement of gratitude is necessary to thesis committee members, Asst. Prof. Kamer Kaya and Asst. Prof. Ayşe Tosun for their presence and valuable feedback. I also owe a debt of gratitude to all instructors in CS department for imparting their knowledge to me.

Special thanks is necessary to my classmates, chatmates and teammates Faizan Suaih and Hamidu Mbonde for their continuous encouragement, mind awakening talks and advises, they always have a special place in my life and require special acknowledgement.

Finally, none of this would have been possible without my family, who has supported and believed me in every situation. I am deeply grateful for their continuous love and support.

CROSS COLLECTION ASPECT BASED OPINION MINING USING TOPIC MODELS

Hemed Hamisi Kaporo

Computer Science and Engineering, Master's Thesis, 2018

Thesis Supervisor: Yücel SAYGIN

Keywords: cross collection topic modeling, aspect based sentiment analysis, text mining.

Abstract

Aspect based opinion mining is the automated science of identifying and extracting sentiments associated to individual aspects in a text document. Over the years this science has emerged to be a cornerstone for analysis of public opinion on consumer products and social-political events. The task is more fruitful and likewise more challenging when comparison of opinion on aspects of multiple entities is of essence. Different methods in literature have attempted to extract aspects in a single collection or collection by collection across multiple collection. These approaches do not appeal when number of collections is large and hence suffer significant performance drawbacks.

In this work we perform aspect based opinion mining across contrasting multiple collections, simultaneously. We utilize existing cross collection topic models to identify topics that prevail across multiple collections, we propose a topic refinement algorithm that successfully converts these topics into semantically coherent and visually identifiable aspects. We compare the quality of aspects extracted by our algorithm to topics returned by two cross collection topic models. Finally we evaluate the accuracy of sentiment scores when measured over features extracted by the two cross collection topic models. We conclude that with proposed improvements cross collection topic models outperform state of art approaches in aspect based sentiment analysis.

KONU MODELLERİNİ KULLANARAK, ÇAPRAZ TEMELLİ GÖRÜŞ MADENCİlİĞİ

Hemed Hamisi Kaporo

Bilgisayar Bilimi ve Mühendisliği, Yüksek Lisans Tezi, 2018

Tez danışmanı: Yücel SAYGIN

Anahtar Kelimeler: çapraz koleksiyon konu modellerini, anlam temelli görüş madenciliği, metin madenciliği.

Özet

Anlam temelli görüş madenciliği, bir metindeki tüm tekil mänäları tanımlayan ve çıkaran otomatikleştirilmiş bilimdir. Yıllar içinde bu bilim, tüketici ürünleri ve sosyal-politik olaylar hakkında kamuoyunun analizinin temel taşı olarak ortaya çıkmıştır. İşin verimliliğiyle beraber zorluk derecesi de birden fazla görüşün değişik anlamlarda farklı kişiler üzerinden araştırılmasıyla artar. Literatürdeki farklı yöntemler tek bir koleksiyonda veya birden fazla koleksiyonda teker teker anlamları bulmayı denemiştir. Bu yaklaşımlar, koleksiyonların sayısı arttığında ve dolayısıyla önemli performans sakıncaları olduğunda cazip değildir.

Bu çalışmada aynı anda birden fazla karşıtlığı da olan koleksiyon üzerinde anlam temelli görüş madenciliği gerçekleştiriyoruz. Birden çok koleksiyonda geçerli olan konuları tanımlamak için mevcut çapraz koleksiyon konu modellerini kullanıyoruz ve bu konuları başarılı bir şekilde semantik olarak uyumlu ve görsel olarak tanımlanabilir anlamlara dönüştüren bir konu iyileştirme algoritması öneriyoruz. Algoritmamız tarafından çıkarılan anlamların başarısını, iki çapraz koleksiyon konu modeliyle dündürülen konularla karşılaştırıyoruz. Son olarak, mn puanlarının doğruluğunu iki çapraz koleksiyon konu modeli tarafından elde edilen özellikler üzerinden ölçerek değerlendiriyoruz. Önerilen geliştirmelerle, çapraz koleksiyon konu modellerinin, anlam temelli mn analizinde son teknoloji yaklaşımlarını geride bıraktığı sonucuna vardık.

Table of Contents

List of Figures

List of Tables

Notations

C = a corpus

c = a collection

d = a document

z = a topic

w = a word

k = number of topics

t = number of aspects

n = number of words representing a topic

$n^{'}$ = number of words representing an aspect

n_d = number of words in a document

n_c = number of words in the collection

n_v = vocabulary size i.e. number of words in the corpus

m = number of documents in the collection

M = number of collections in the corpus

$\theta = p(z_j)$ for all js i.e. θ is a vector

$\pi = p(w_i|z_j)$ for all i(s) and j(s) i.e. π is a vector

λ_B = probability of selecting a background(stop) words distribution

λ_z = probability of selecting a topic words distribution

$c(w_i, d)$ = number of occurrence of word w_i in document d

Chapter 1

Introduction

Opinion mining also referred to as sentiment analysis is the science of extracting public opinion towards products or events from unstructured text. Most people express their opinions on social events and consumer products in plain unstructured text in social media (social networks and review forums). This makes unstructured text the major source of public opinion. Needless to say, public opinion on social issues and consumer products is of paramount importance as they can shape societies, affect product sales or influence political elections.

An opinion as defined by Bing Lu et al. [1] is a quintuple of entity, aspect of the entity, opinion orientation of the aspect and the time an opinion was given. From this definition opinion mining can be formalized as the task of identifying a set of quintuples given an opinionated text.

Extraction of quintuples can be done at different levels of the opinionated text. Given an opinionated text (e.g. a tweet or a product review), document-level opinion mining aims at classifying the whole text as being positive, negative or neutral towards a particular entity. Sentence-level opinion mining aims at finding opinion orientation of every sentence in the opinionated text. For both levels classification can be turned into regression to express opinion orientation in a wider range of values.

Mining of opinionated texts at the document level or at the sentence level is useful but has some notable downsides. A positive opinionated document about a particular entity hardly implies that the author has positive opinions on all aspects of the entity. Likewise, a negative opinionated document does not mean that the author dislikes everything. Although the general sentiment on the entity may be positive or negative, an author of a typical opinionated document expresses negativity on some aspects of the entity while remains positive on others. Document and sentence-level sentiment analysis does not provide such information. To obtain these details, aspect-level sentiment analysis is introduced.

Aspect based opinion mining aims at finding opinion orientation of every aspect of an entity in the given opinionated text. This means finding a full set of quintuples without any relaxation. Given a collection of reviews on a product, say phoneA, aspect based opinion mining returns opinions specific to each of phoneA's aspect such as battery life, performance, memory, size and camera quality. This task requires two major steps; first is the extraction of aspects of phoneA from the given review collection and second is to map these aspects to their respective opinion words or phrases. Intense research and publications have been done on methods of extracting aspects and mapping aspects to opinion orientations. While [2–13] used topic models to extract product aspects, [14–16] used hidden markov models. Other methods include usage of conditional random fields [17, 18] or traditional parts of speech (POS) tagging [19].

Although all these methods perform well in practise, they only appeal to the extraction of aspects in a single collection. less effort is directed to situations where a comparison of multiple products (entities) is of essence. [16, 20] made an attempt to compare peoples' opinions on aspects of multiple products. The approach in both works involve running aspect based opinion mining algorithms to each product collection separately and later merge results for presentation. This approach is not practical; space and computational complexity increases linearly with the increase in number of collections (assuming one entity per collection) to compare. Another possible solution is merging reviews of multiple products in a single collection and running aspect based opinion mining algorithms to the merged collection. This attempt is not effective because users usually do not explicitly mention the entity name in each review, mixing reviews of multiple entities on a single collection makes attribution of opinions and aspects to right entities an impossible task.

In our work we use cross-collection topic models to perform aspect-based sentiment analysis for multiple entities simultaneously. We base our discussion on cross collection Latent Dirichlet Allocation (ccLDA) and Cross Perspective Topic Model

(CPTM) which are cross collection topic models proposed in [21] and [22] respectively. While the former extracts topics that are common to all collections and topics that are independent to each collection, the later uses nouns to model collection-independent topics and adjectives, adverbs and verbs to model collection-specific topics. The arrangement in both opens a new possibility of modelling collection independent topics as aspects and corresponding collection specific topics as opinion towards these aspects. We therefore propose a topic refinement algorithm based on coherent cluster growth and word vectors to convert topics returned by these models into identifiable aspects. We argue that with this refinement, topics elicited from these models align perfectly with product aspects. To this end we compare the intrinsic semantic topic coherence of aspects refined by our method to topics returned directly from ccLDA and CPTM. We show that our approach outperforms both.

Although ccLDA and CPTM uses bag of words to model topics, which means no semantic relation can be inferred from elicited topics, we investigate the role of collection specific topics in ccLDA (also known as opinion topics in CPTM) in determining sentiment scores of elicited aspects.

1.1 Thesis Motivation

This thesis is motivated by the desire to contribute to efficient aspects-wise comparison of public opinion on multiple entities. The problem was first defined by ChengXiang zhai et al. [23] as comparative text mining problem. This initial work and subsequent attempts such as that of Michael Paul et al. [21] and Yi Fang et al. [22] propose powerful topic models that model common topics across multiple collections and topics specific to each collection.

Although these models are useful for topic extraction, they do not guarantee semantic intepretability of the topics. A bag of words is used to model topics. Thus, resulted topic-words do not necessary demonstrate semantic coherence. Moreover, these models are not particularly suited for opinion mining. For example, a topic in [23] and [21] not only contains words that describe the topic, but also words that express opinions about the topic. This makes opinions and topics obscure for opinion mining. [22] modelled opinion words and topic words separately but did not propose any method to obtain numeric scores of the elicited opinion words and phrases. We therefore intend to improve these models to enable multi-entity multi-aspect sentiment analysis.

1.2 Thesis Contribution

In this work, we combine cross collection topic modelling and sentiment analysis to form a framework that performs cross collection aspect based sentiment analysis. We propose a topic refinement algorithm based on coherent cluster growth and word vectors to produce highly semantically coherent and visually identifiable aspects. We argue that with this refinement, topics elicited from cross-collection topic models align perfectly with product aspects. Finally we perform lexicon based sentiment analysis using opinion words extracted from these models as features. To this end we conclude that the use of such features for sentiment analysis yields more accurate sentiment scores than supervised counterparts

Chapter 2

Related work and Preliminaries

Aspect based opinion mining is a two phase problem, first is the extraction of aspects from the given corpus and second is the association of the extracted aspects to opinions that well represent the given data. In this chapter, we examine previous works related to these two portions of the problem. We first introduce topic modeling and examine its historical development towards aspect discovery in multiple data collections. Then we explore opinion extraction methods. Finally we briefly introduce related background knowledge such as parameter estimation and model evaluation techniques.

2.1 Topic Modeling

Topic models aim at finding latent(hidden) structures in a collection or multiple collections of data, In recent years topic models have proven to be successful in identifying hidden structures in textual data [21–27], image data [28] and medical data [29–33]. Definition of a hidden structure depends highly on the data in question. For text data latent structures have been defined to be underling topics in the given corpus [21–27], product aspects [2–13] or query words [34]. In this section we focus on text data, we stick to the traditional definition of latent structures being underlying topics in the given corpus. In later sections the notion of hidden structures is extended to mean product features.

Several techniques have been used to discover these hidden structures, although [24] used linear algebra and matrix decomposition, the majority of literature defines topic modeling as a probabilistic modeling problem. Probabilistic topic modeling is characterized by two main sub-problems; first is defining the generative process of a document, and second is the problem of parameter inference. A generative process explains how words in a document might be generated on the basis of random variables. Given a document collection parameter inference tries to find the set of latent variables that best explains the observed data.

Probabilistic topic models mainly differ in the assumptions put forward to define the document generative process. A change in statistical inference algorithm does not necessary alter the model identity. Scope of the model is another important aspect. Some models only handle a single collection while others span across multiple collections. With each change in document degenerative process or data scope a new topic model is born.

In this section we examine the historical development of topic models. The section starts with models that operate on a single collection then extends the subject to cross collection topic models.

2.1.1 Single Collection Topic Models

The simplest probabilistic topic model is presumably the unigram model. This model assumes that there is only one topic in a document collection, i.e. a document is generated by drawing each word independently from a single multinomial distribution of words. Probability of a document is therefore given by.

$$p(d) = \prod p(w_i) \tag{2.1}$$

Since topic models output topics as multinomial distribution of words, then given a collection, unigram model tries to infer the distribution P(w).

By extending unigram model to in-cooperate multiple topics, the mixture of unigrams (MU) model [35] is formed. MU assumes that a collection expresses multiple topics, with each document exhibiting only one of these topics. Probability of a document is therefore given by.

$$p(d) = \sum_{j=1}^{k} p(z_j) \prod_{i=1}^{n} p(w_i|z_j) \tag{2.2}$$

i.e, a topic that the document should express is chosen with probability $p(z_j)$

and then each word is included in that document with probability $p(w_i|z_j)$. Thus, given a collection, MU tries to infer the topics distribution θ, and the topic-words distribution π. Probability inference for mixture models is an ancient problem with multiple solutions in literature. Algorithms like expectation maximization (EM), Gibbs Sampling, variational inference and particle filtering attempt to solve this problem. In chapter 3 a brief overview of these algorithms is given.

Unigram model and mixture of unigrams model form a baseline for topic modeling and they usually hold true for short documents like tweets where each document virtually addresses only a single topic. Unfortunately these models fail to capture the pivotal reality for long documents where a document usually expresses multiple topics. To address this issue Thomas Hoffman introduced probabilistic latent semantic analysis (PLSA) [25].

PLSA assumes there are multiple topics in a collection, each document is a distribution of topics and each topic is a distribution of words. No assumption is made on the type of these distributions. A word in a document is generated by first choosing a topic it represents by $p(z_j|d)$. Then draw the word from that topic distribution by probability $p(w_i|z_j)$. These choices are made for every single word in the document. Thus, probability of a document is given by.

$$p(d) = \prod_{i=1}^{n_d} \sum_{j=1}^{k} p(z_j|d) p(w_i|z_j) \tag{2.3}$$

Given a document, PLSA aims at finding $p(z_j|d)$ for every j and $p(w_i|z_j)$ for very i, constrained at $\sum_{j=1}^{k} p(z_j|d) = 1$ and $\sum_{i=1}^{n_d} p(w_i|z_j) = 1$. $p(z_j|d)$ for every j is referred to as the θ vector and $p(w_i|z_j)$ for every i is referred to as the π vector. θ and π can then be given by equation 2.4.

$$p(\theta, \pi|d) = \frac{p(\theta, \pi, d)}{p(d)} \tag{2.4}$$

This is a posterior inference problem. To solve this problem, PLSA uses Expectation Maximization (EM) algorithm, which aims at finding θ and π that maximizes the log likelihood of the document collection.

Although PLSA's generative process captures the reality of many documents in practise, it falls short in parameter estimation. No assumption is made about the distribution governing θ and π, independent probabilities θ and π have to be determined for every word and topic in the collection. This results to a huge number of parameters to be estimated. Another issue is that PLSA is not a well defined generative model of documents, this is because it only tries to fit the training document set. It conditions probability of a word and that of a topic to a specific document. In other words PLSA tries to learn the topic mixtures θ only for those documents on which it is trained. Due to this fact it can not be used to determine topic proportions of an unseen document. To tackle these problems David Blei et al. introduced the famous Latent Dirichlet Allocation (LDA) [26].

LDA proposes nearly the same document generative process as that of PLSA. Its main addition is the definition of probability distribution from which topics and words originate. It assumes that each document is a multinomial distribution of topics and each topic is a multinomial distribution over words. Explicit definition of these distributions makes it possible for observers to insert their prior knowledge. Dirichlet distribution is a conjugate prior to multinomial distribution, therefore by adding dirichlet prior α to topic proportions θ and dirichlet prior β to topic-words distribution π, LDA extends PLSA.

Intuitively given a collection, say documents on world history, LDA allows a reader to insert his or her prior knowledge about topic proportion in the documents. For instance, a collection on world history is know to have large topic portion on ancient history, moderate in medieval events, and fairly small writings on modern era.

Given a document and prior knowledge α and β, LDA aims at finding θ and π. This can be represented as;

$$p(\theta, \pi | d, \alpha, \beta) = \frac{p(\theta, \pi, d | \alpha, \beta)}{p(d | \alpha, \beta)} \tag{2.5}$$

Where:

$$p(\theta, \pi, d | \alpha, \beta) = p(\theta | \alpha) \prod_{i=1}^{n_d} p(z_j | \theta) p(w_i | z_j, \beta) \tag{2.6}$$

And

$$p(d | \alpha, \beta) = \int p(\theta | \alpha) \Big(\prod_{i=1}^{n_d} \sum_{j=1}^{k} p(z_j | \theta) p(w_i | z_j) \Big) d\theta \tag{2.7}$$

The above posterior inference is computationally intractable for exact inference, the marginal probability is a multiple hyper-geometric function or a sum of n^k

Dirichlet integral terms. For this reason LDA settles for approximate posterior inference. Methods for approximate posterior inference include Gibbs sampling, variational inference and particle filtering to be discussed in chapter 3.

Since its introduction several topic models have been proposed to extend LDA. For instance, LDA upholds the bag of words model, i.e. LDA assumes that, the order of words in a document is of no significance, only number of occurrence of these words is of essence. This assumption is an over simplification of the true nature of documents. To address this issue, Wallach et al. [36] proposed a model that eliminates the exchangeability assumption by assuming that a word is generated by a topic depending on its previous word.

Another issue is that LDA do not model correlation between topics, it only models correlation between words. Given a set of words, LDA detects whether the words correlate (fall under the same topic) or not, contrary to that, given a set of topics, LDA suggests no method to determine whether these topics correlate (are subtopics of one larger topic) or not. To address this issue several hierarchical topic models have been proposed. These models include Correlated Topic Model [37], Pachinko Allocation [27] and Hierarchical Latent Dirichlet Allocation [38].

In this section we discussed the evolution of single collection topic models. Given a collection of unlabeled documents these models try to discover underlying topics of the collection. Several other models have been proposed to try to learn topics from a set of labeled documents. In supervised topic models [39–48], documents are given labels such as number of likes associated with a document, the task of the model becomes to predict number of likes in unseen document based on similarities or differences of topic proportion between the labeled and unlabeled document. In the next section we discuss another family of topic models. Topic models designed to operate on multiple document collections.

2.1.2 Cross Collection Topic Models

Cross collection topic models aims at finding common topics across all comparable collections and topics that are unique to each collection.

The first cross collection topic model was proposed by ChengXiang zhai et al. [23] in an attempt to solve the problem they so defined as a Comparative text mining problem. The model was named as cross collection Mixture model (ccMix). This model is a direct extension of PLSA to accommodate multiple collections. It assumes that a document in a multicollection corpus contains topics(themes) that are only specific to its collection and themes that are common to all collections. ccMix aims at extracting what is common to all collections and what is unique to one specific collection. Probability of a document as proposed by ccMix is therefor given by:

$$p(d) = \prod_{i=1}^{n_d} \sum_{j=1}^{k} p(z_j) \Big(\lambda_c p(w_i|z_j) + (1 - \lambda_c) p(w_i|z_j, c) \Big) \tag{2.8}$$

Where: λ_c is the probability of drawing a word from the collection independent word distribution.

Due to the fact that ccMix uses the PLSA way of thinking, it faces the same problems as that of PLSA. No assumption is made about the distributions governing the topic proportions in a document nor to collection-independent or collection-specific word distributions. This results to a huge number of parameters to be estimated. Furthermore ccMix fails to generalize and hence can not be used to determine topic proportions of unseen documents.

As it was for single collection topic models, a better alternative to ccMix that addresses all these issues was introduced. The model is named cross collection Latent Dirichlet Allocation (ccLDA) and it replaces the PLSA backbone of ccMix to that of LDA. In other words ccLDA to ccMix is as LDA is to PLSA.

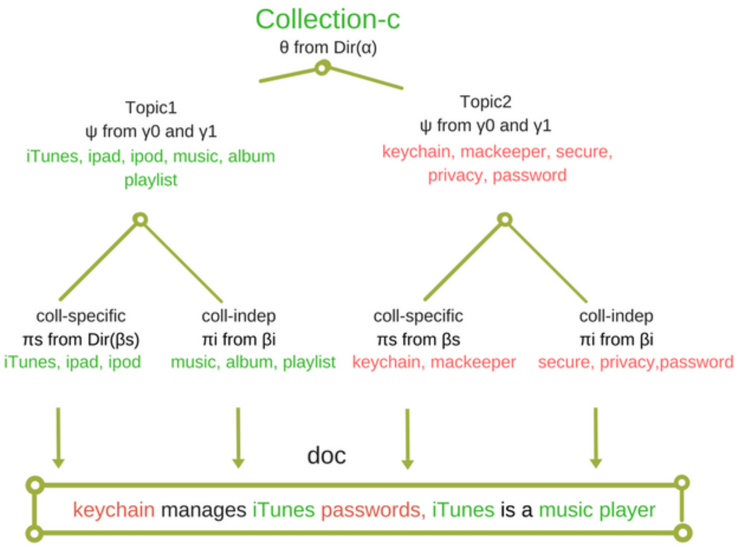


Figure 2.1: Overview of ccLDA's document generative process

Figure 2.1 shows the ccLDA document generative process with an example document. The generative process of a document follows two steps; first is the generation of word distributions and topic proportions while the second step is picking words from these distributions to make the document.

In the first step ccLDA samples a collection c out of multiple collections. Then, samples multinomial topic proportions θ from Dirichlet α for documents in the collection. Since each topic is assumed to contain words from either collection-independent words distribution or collection specific words distribution, then ccLDA samples Bernoulli proportion ψ of these distributions from Beta(γ_0, γ_1). Note, γ_0 encodes information belonging to a collection-independent words distribution while γ_1 encodes information belonging to a collection-specific words distribution i.e. if γ_0 is set greater than γ_1 then a topic is assumed to have more words from collection-independent multinomial words distribution than from collection specific words distribution. Finally, ccLDA draws collection-independent multinomial words distribution π_i from dirichlet β_i for each topic and collection-specific multinomial word distribution π_s from dirichlet β_s for each topic and collection.

To add a word in a document in collection c, ccLDA decides on a topic z from θ to pick the word from. According to ψ of that topic ccLDA goes on to decide either to draw the word from collection specific or collection independent distributions. Finally a word is drawn from π_i or π_s accordingly.

As stated earlier, exact inference is often intractable for complex Bayesian models, ccLDA is no exception. Gibbs sampling is therefor used for approximate inference of ccLDA.

Although ccLDA performs well in practise for comparing topics in multiple collections it does not model opinions on these topics. A topic in ccLDA not only contains topic words, but also words that express opinions about the topic. In other words, ccLDA does not differentiate opinion words from topic words, which makes both opinions and topics obscure for opinion mining. To solve this problem a cross perspective topic model (CPTM) [22] is introduced.

CPTM assumes that, opinion generation process is separated from the topic generation process. This makes CPTM the best model for cross collection opinion mining. Figure 2.2 shows the generative process of a document as proposed by CPTM.

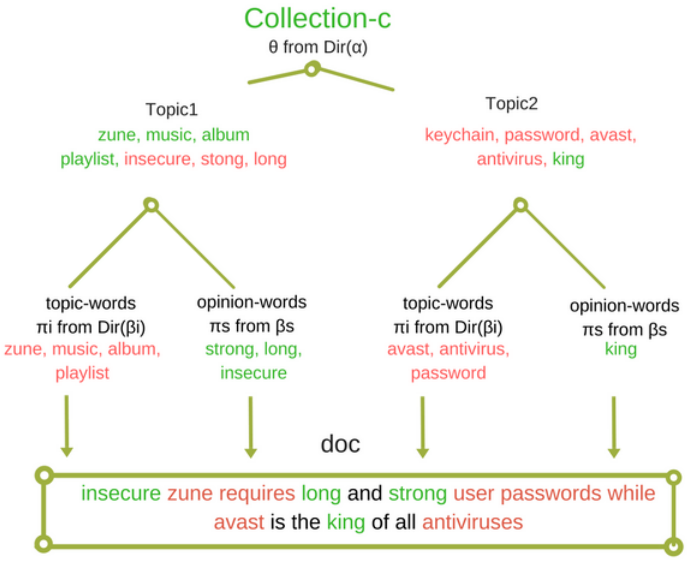


Figure 2.2: Overview of CPTM's document generative process

As in other models, the generative process of a document in CPTM involves two steps; the first step is the generation of word distributions and topic proportions while the second step is picking of words from these distributions and adding them to the document. To generate distributions, CPTM first samples a collection c out of multiple collections. Then, samples multinomial topic proportions θ from Dirichlet α for documents in the collection. CPTM then draws a collection-independent multinomial topic-words distribution π_i from dirichlet β_i for each topic and collection-specific multinomial opinion-words distribution π_s from dirichlet β_s for each topic and collection.

To add a topic word (noun) in the document, CPTM decides on a topic z from θ to pick the word from and draws the actual word from π_i of that topic. To add an opinion word i.e. adjective, adverb or verb to the document CPTM again decides on a topic z from θ to pick it from and draws the actual word from π_s of that topic in that particular collection.

The last cross collection topic model of interest is Topic Aspect Model (TAM) [49] introduced by Michael Paul et al.

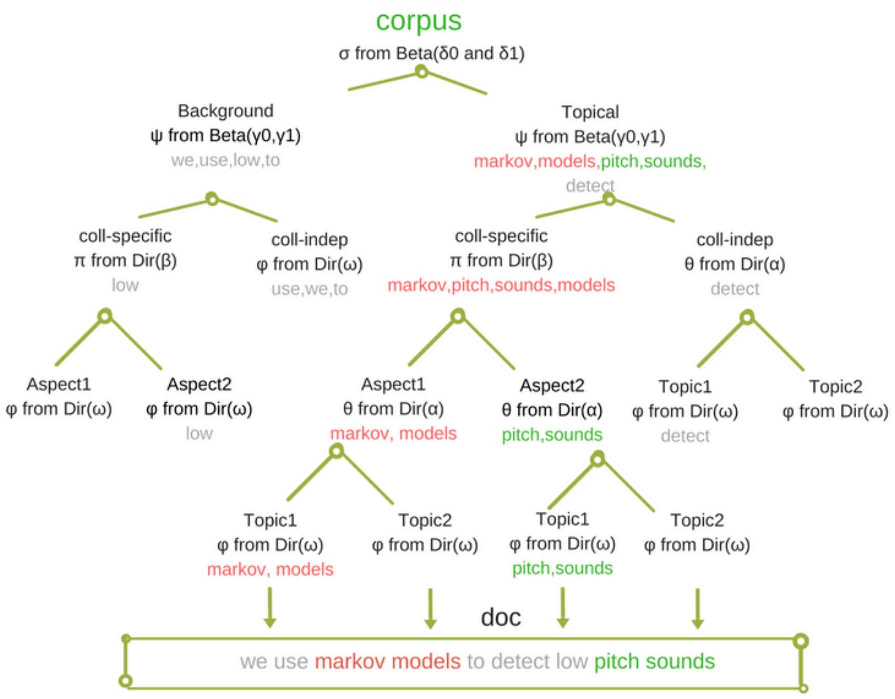


Figure 2.3: Overview of TAM's document generative process

The main difference between TAM and its counterparts i.e. ccLDA and CPTM is its assumption that not only a document contains words from multiple topics but also from multiple themes (perspectives). The document generative process of TAM is as shown in figure 2.3. Using a computational linguistics paper as an example Michael Paul explains that the paper may contain computational terminologies such as algorithms, models etc. as well as linguistic terminologies such as language, semantics, pitch etc. This is different from ccLDA and CPTM where there would be two collections, a linguistic and a computational collection, each with documents containing words specific to that collection and words common to both collections.

Another difference is TAM's ability to model background/stop words. ccLDA and CPTM assumes that stop words are eliminated before model execution, these models only deal with words that are topical i.e., convey a certain meaningful information, on the other hand TAM models existence of stop words in the corpus.

2.2 Sentiment Classification

In section 2.1 we examined topic modeling as a method of extracting aspects from a given document or collection(s). Some cross collection topic models such as CPTM went a step further to even extract opinion words and phrases associated to these aspects. In this section we examine methods that can be used to quantify opinions associated to the extracted aspects.

Classification based methods

This is the supervised or semi-supervised method to sentiment analysis where the problem is posed as a binary classification problem [50–53]. Aspects, sentences or documents are assigned a binary sentiment value i.e. either positive or negative. Pang and Lee [54] showed that sentiment classification can be generalized into a rating scale, this qualifies the problem as a regression problem. The intuition is that, one gets better diversification of sentiments when using a rating scale than when binary classification is used.

Lexicon/Dictionary based methods

This is the use of opinion lexicons i.e. a list of opinion words and phrases, and a set of rules to determine opinion orientation of aspects in a document. Although the classification based approach is the dominant approach towards sentiment analysis in literature, Sagar Ahire in his survey of sentiment lexicons [55] pointed out that sentiment analysis is different from text classification and therefore not as suited for machine learning techniques.

There exist many lexicons in literature, most popular are the Affective Norms for English Words (ANEW) [56] and sentiwordnet. Unlike sentiwordnet that contain only a single sentiment score per word, ANEW contains scores for three sentiment categories; valence, arousal and dominance. Valence score attests the polarity of a word ranging from negative to positive, arousal indicates the excitement level ranging from highly excited to calm while dominance reflects how certain the user is in expressing the sentiments.

2.3 Parameter estimation and inference

Topic models are a form of mixture models i.e. words from a document are drawn from multiple topic distributions. Analytical parameter inference methods such maximum likelihood (ML) and Maximum posterior (MAP) estimates become impossible for these complex models. For that reason iterative methods such as expectation maximization (EM), sampling methods and variational based methods are employed. In this section we briefly examine how these iterative methods are used to infer model parameters in topic modeling context.

2.3.1 Expectation Maximization

Expectation Maximization (EM) [57] algorithm is an iterative method that seeks to find maximum likelihood (ML) or maximum aposterior (MAP) estimates of parameters in statistical models. Using PLSA, the mother of probabilistic topic models we briefly explain how EM is used to infer parameters θ and π.

Looking at PLSA as a mixture model i.e. words in a document come from multiple topics. EM intuitively determines the topic assignment for each word, which in turn makes solving for θ and π easy. EM follows two basic steps, the expectation and the maximization step as explained below.

Expectation (E) - step:
In this step, EM algorithm computes expectation of each word to belong to a specific topic i.e. $p(z_j|w_i, d_q)$ and assigns a word to the topic it is highly expected to fall under i.e. a topic with highest $p(z_j|w_i, d_q)$.

Note:

$$p(z_j|w_i, d_q) = \frac{p(z_j)p(w_i, d_q|z_j)}{p(w_i)} = \frac{p(z_j)p(w_i|z_j)p(d_q|z_j)}{\sum_{j=1}^{k} p(z_j)p(w_i|z_j)p(d_q|z_j)} \tag{2.9}$$

Where: $p(z_j)$, $p(w_i|z_j)$ and $p(d_q|z_j)$ are randomly initialized and expected to be updated in subsequent iterations.

Maximization (M) - step:

This step aims at updating model parameters i.e. $p(z_j)$, $p(w_i|z_j)$ and $p(d_q|z_j)$ such that when used in the E-step, expectations are maximized. Parameters $p(z_j)$, $p(w_i|z_j)$ and $p(d_q|z_j)$ are given by.

$$p(z_j) = \frac{\sum_{q=1}^{m} \sum_{i=1}^{n_d} c(w_i, d_q) p(z_j|w_i, d_q)}{\sum_{j=1}^{k} \sum_{i=1}^{n_d} c(w_i, d) p(z_j|w_i)} \tag{2.10}$$

$$p(w_i|z_j) = \frac{\sum_{q=1}^{m} c(w_i, d_q) p(z_j|w_i, d_q)}{\sum_{i=1}^{n_v} \sum_{q=1}^{m} c(w_i, d_q) p(z_j|w_i, d_q)} \tag{2.11}$$

$$p(d_q|z_j) = \frac{\sum_{i=1}^{n_d} c(w_i, d_q) p(z_j|w_i, d_q)}{\sum_{i=1}^{n_v} \sum_{q=1}^{m} c(w_i, d_q) p(z_j|w_i, d_q)} \tag{2.12}$$

E-step and M-step are repeated till convergence. Convergence is when no notable changes are observed in the values $p(z_j|w_i, d_q)$. EM algorithm is guaranteed to converge but not necessary to global maximum [58], it has been attributed to stacking at local maximum in some applications. For this reason, other methods such as Sampling methods are preferred.

2.3.2 Sampling Methods

Instead of trying to compute posterior model parameters, Sampling based methods try to recreate the posterior distribution, From the created posterior distribution model parameters can then easily be estimated. These methods include rejection sampling, importance sampling and Markov Chain Monte Carlo (MCMC) based methods. For convenience we briefly examine Markov Chain Monte Carlo methods.

Markov Chain Monte Carlo (MCMC)

This is the family of methods used to estimate model parameters in the two topic models of interest. A Monte Carlo algorithm is the one that estimates properties of a distribution based on a large number of samples from the given distribution. Markov chain is the idea that samples are generated by a special sequential process. Each random sample is used as a stepping stone to generate the next random sample (hence the chain), each new sample depends only on the one before it. New samples do not depend on any sample before the previous one (this is the Markov property). MCMC is a family of methods, the famous ones include Metropolis Hastings, Gibbs sampling and their variations.

2.4 Evaluation Measures

Several methods have been introduce in an attempt to measure the performance of a topic model. Contrary to speed and space algorithmic complexity measures, this section focuses on quality measures of topic models. The measures include how well the extracted topic-words are semantically coherent, how well topics are understandable, whether the returned topics encompass all topics available in the corpus i.e. no topic is left out, and whether the model can generalize well to unseen documents.

2.4.1 Perplexity

Perplexity is a quantitative measure for comparing language models and is often used to compare the predictive performance of topic models [59]. The value of perplexity reflects the ability of a model to generalize to unseen data. A lower perplexity score indicates better generalization performance. To measure perplexity of a topic model, a collection is divided into train and test sets of documents. The model is run on the training documents set and the discovered topics are tested in the test set as in equation 2.13.

$$H(D') = \sum_{D'} p(w_d) log_2 p(w_d) \tag{2.13}$$

$$Perplexity = 2^{H(D')} \tag{2.14}$$

Where H(D') is the held-out likelihood of (test) documents D'. Recent studies have argued that perplexity is not a better topic modeling evaluation measure, [60] have shown that predictive likelihood i.e. perplexity and human judgment are often not correlated, and sometimes slightly anti-correlated.

2.4.2 Word Intrusion

This is a qualitative intrinsic evaluation method of topic models proposed by Chang el al. [60]. It is mainly introduced to correct the shortcomings of perplexity measure. It measures how well top words in extracted topics are semantically coherent. It does so by introducing a top word from one of the topics into another topic.

A human observer is then asked to identify which word among words in the latter topic seems out of place. The process is repeated to all topics and using multiple human observers. The average response on each topic is then recorded. A model is said to have high topic coherence and hence a good model if intruder words could easy be identified.

The main shortcoming of this measure is the fact that it requires human annotators, hence it is subjected to bias and not suitable for evaluation of large datasets.

2.4.3 Topic coherence

The notion of topic coherence also referred to as confirmation measure was first introduced by Newman et al. [61]. Confirmation measure is a family of measures aiming at automatically evaluating topic models without the help of human annotators. The methods compute pairwise similarity between topic words and aggregate the similarity measure to obtain the confirmation measure of a topic.

The methods differ on the similarity measures used. Different similarity measures have been used in literature, among them is pairwise mutual information (PMI) [61], normalized PMI [62] and log conditional probability (LCP) [63] as described below.

$$PMI(t) = \sum_{i=2}^{N} \sum_{j=1}^{i-1} log \frac{p(w_i, w_j)}{p(w_i)p(w_j)} \tag{2.15}$$

$$NPMI(t) = \sum_{i=2}^{N} \sum_{j=1}^{i-1} log \frac{\frac{p(w_i, w_j)}{p(w_i)p(w_j)}}{-p(w_i, w_j)} \tag{2.16}$$

$$LCP(t) = \sum_{i=2}^{N} \sum_{j=1}^{i-1} log \frac{p(w_i, w_j)}{p(w_j)} \tag{2.17}$$

Where $p(w_i, w_j)$ is the ratio of number of documents in the held-out (test) set containing both word w_i and w_j to the total number of documents in the test set. $p(w_j)$ is the ratio of number of documents in the test set containing word w_j to the total number of documents in the test set. N is the number of top words in a topic, t.

PMI ranges from 0 to $+\infty$ where $+\infty$ indicates strong correlation between topic words and 0 indicates no correlation. NPMI was introduced to normalize pmi value form -1 to 1. 1 being highly correlated.

Chapter 3

Methodology

In this chapter we explain and discuss our method for cross collection aspect based opinion mining. We start by exploring the nature of datasets that we deploy in experimentation and evaluation. In section 3.4 we reveal our topic refinement algorithm. Lastly in section 3.5 we perform sentiment analysis on the extracted aspects based on opinion words generated from the topic models.

3.1 Data Sets

To achieve fair judgment and decent evaluation of the two topic modelling algorithms, we test the models in three data categories. The first two categories consist of two datasets each while the last category is made up of a single dataset making a sum of five datasets for experimentation and evaluation. The three data categories are as follows:-

1. **Short documents, large collections**. Short documents usually express opinion on a single aspect per document. By examining datasets in this category we hope to capture the performance of the two topic models in the most fundamental task of identifying an aspect in a document. Social media data is a good example of this category where users convey their opinions in the form of short documents. Large number of documents per incident (collection) can also easily be accessed due to the large number of people contributing their opinions in social media. To mimic this data category we made use of two datasets from Twitter; Airlines dataset and Debate dataset. Length of a stan-

dard tweet is 140 characters, this is considered short relative to other data sources such as news sites and review forums.

- **Airlines dataset:** This is a dataset downloaded from kaggle[1]. The dataset consist of tweets targeting five major United States airlines in February 2015. Airline users through Twitter comment on issues such as quality of customer services, flight delays, on board comfortability, costs, and other airlines related aspects for the five airlines. The Tweets are location tagged as US and Canada. The distribution of number of documents(tweets) per collection(airline) is as illustrated in Table 3.1.

Table 3.1: Airlines dataset

Airline	Number of documents
American	2724
Delta	2165
Southwest	2362
United	3874
UsAirways	2823

- **Debate dataset:** This is a dataset downloaded from kaggle[2]. The dataset consist of tweets targeting major party candidates for the 2016 United States presidential election i.e Donald Trump and Hillary Clinton. These tweets are mainly comprised of peoples' views on policies put forward by these two candidates, their work ethics, experience and personal life. The dataset initially consisted of 8448 tweets. After removing retweets and separating those tagged realDonaldTrump from HillaryClinton we remained with two collections. Collection Donald Trump and collection Hillary Clinton. The Donald Trump's collection has a sum of 3903 documents(tweets) while Hillary Clinton's collection has 3678 documents.

[1] https://www.kaggle.com/crowdflower/Twitter-airline-sentiment

[2] https://www.kaggle.com/benhamner/clinton-trump-tweets

2. **Long documents, small collections**. In long documents people usually comment on multiple aspects of an incident or product. Small collection size means weak aspects emphasis i.e. an aspect is mentioned in few documents. By including datasets of this category we intend to investigate how well the three topic models extract all aspects in a particular document, even when the aspects are not well emphasized throughout the collection. We make use of movies dataset the hotels dataset.

   - **Movies dataset:** This dataset is a subset of amazon movies review dataset [3]. The subset contains reviews on five popular movies of the 21^{st} century. The number of documents per collection (movie) is as illustrated in Table 3.2.

Table 3.2: Movies dataset

Movie(year)	Number of documents
The notebook(2004)	777
Alexander(2004)	659
Apocalypto(2006)	589
Gran torino(2008)	531
The best of schoolhouse rock(1998)	589

   - **Hotels dataset:** This dataset consists of user reviews from tripadvisor.com[4] on three hotels; Rio mar in Puerto Rico; Iberostar and Caribe club princess in Dominican Rep. Number of documents (reviews) for each collection (hotel) is around 500, where each review contains 500 words or more. Table 3.3. shows distribution of number of reviews per collection in the hotels dataset.

Table 3.3: Hotels dataset

Hotel	Number of documents
Rio mar (Puerto Rico)	610
Iberostar (Dominican Rep.)	536
Caribe club princess (Dominican Rep.)	543

[3] http://snap.stanford.edu/data/web-Movies.html

[4] https://www.tripadvisor.com

3. **Long documents, large collections**. Contrary to the previous category where the collection size is smaller, aspect emphasis is high in large collections. As any other learning algorithms, topic models are expected to perform well when there is abundance of data. Therefore, by including this category we intent to investigate the best case scenario for the two topic models in the task of aspect extraction. We make use of the cell phones dataset.

    - **Cell phones dataset:** This dataset consists of scrapped user reviews from gsmarena[5] on five cell phones. Due to the wide spread usage of cell phones it was possible to collect large collections of cell phone reviews. Table 3.4 shows distribution of number of reviews per collection in the cell phones dataset.

Table 3.4: Cell phones dataset

cell Phone	Number of documents
samsung galaxy note 7	10214
blackberry curve 9320	5931
htc one m7	7383
iphone6	8022
sony xperia xz	8022

[5] https://www.gsmarena.com

3.2 Preprocessing

Input to ccLDA and CPTM is similar with minor differences. Both algorithms take a corpus of documents at a time, documents are grouped by their collection of origin, see figure 3.1. To mimic this input arrangement we represent a document as a file and a collection as a directory. After achieving this arrangement using physical files and directories we tokenize the documents and remove stop words and punctuations using the standard list of stop words[6]. We further remove web links and hash-tags for Twitter documents and we convert all words to a common case (lower case) to maintain word consistence. We however avoid lemmatization so that to capture negative and positive form of a word as different words rather than same words.

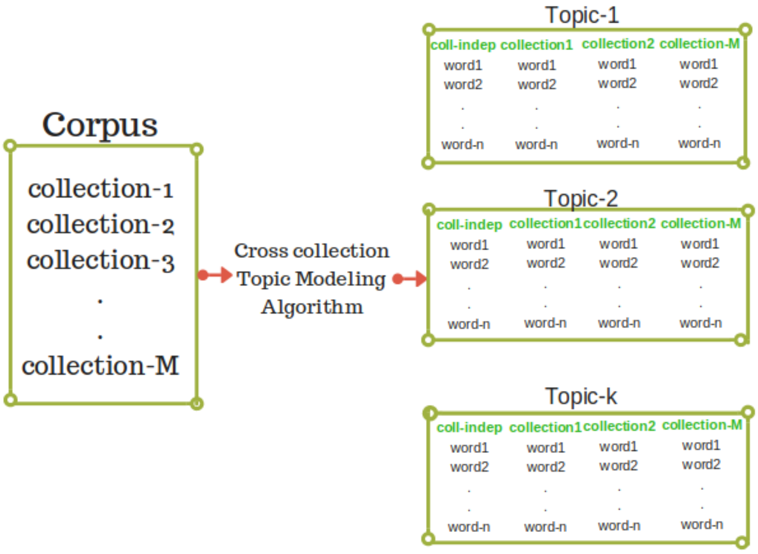


Figure 3.1: Input and output of a cross collection topic model

After common preprocessing we perform algorithm-specific preprocessing. Before feeding each corpus to CPTM we use parts of speech tagging to identify nouns. We then separate nouns from other word types in each document. This step in necessary due to the fact that CPTM treats topic words (nouns) and opinion words (adjective, verbs and adverbs) differently as explained in section 2.1.2. For ccLDA we do not perform this step. This is because ccLDA treat words of all types similarly as explained in section 2.1.2.

[6] http://www.nltk.org/nltk_data/

3.3 Topic extraction

In this subsection we elucidate the algorithmic environment that we set when running algorithms for topic extraction. We extract topics using two topic modeling algorithms. The first algorithm is ccLDA which is based on standard LDA as described in section 2.1.2. This will act as a baseline for topic extraction. Second we extract topics using CPTM which is basically LDA executed over nouns only. In later sections we try to refine the extracted topics to attain visually identifiable aspects.

3.3.1 Cross Perspective Topic Model (CPTM)

As explained in section 2.1.2, CPTM has two corpus-wide hyper parameters α and β; and collection specific hyper parameters β_i for each collection. The work by T. Griffiths et al. [59] shows that these hyper parameters only affect the convergence of Gibbs sampler but not much the output results. For this reason we fix $\alpha = 50/\mathrm{k}$ where k is the number of topics and $\beta = \beta_i = 0.02$ for all i's in all experiments as suggested in the original CPTM work. We set k as 40 and run the algorithm for 200 iteration. We finally request an output of 10 words per topic.

3.3.2 Cross Collection Latent Dirichlet Allocation (ccLDA)

ccLDA has five hyper parameters α, γ_0, γ_1, β_c and β_s as explained in section 2.1.2. We fix these parameters at $\alpha = \gamma_0 = \gamma_1 = 1.0$ and $\beta_c = \beta_s = 0.01$ for every experiment. We run the algorithm for 3000 iterations with number of topics, k = 40. We finally request an output of 10 words per topic.

Note: The number of topics k, for both ccLDA and CPTM is chosen based on prior knowledge of the dataset, It is known that number of distinct topics to be extracted in any of the five datasets is less than 40. Choosing k less than 40 results to high perplexity values, which is an indication of poor generalization while any number above forty is unrealistic based on the prior knowledge of the dataset.

3.4 Postprocessing

Output of the topic extraction task in the previous section is a list of 40 topics for each dataset. Each topic contains collection independent and collection specific words arranged accordingly. See appendix A for summary of these results.

The summary of ccLDA and CPTM results in appendix A shows that, extracted topics barely represent one coherent concept. This is because both ccLDA and CPTM are based on LDA which in-turn uses bag of words to model topics. This means topic-words are not conditioned to display semantic coherence. For example, topic26 returned by CPTM is represented by words $\{room, night, dinner, breakfast, nights\}$. It is hard to conclude whether the described concept is a **room** or **food**. This section aims at converting these topics into well defined, visually identifiable aspects.

3.4.1 Topic refinement

Topic refinement algorithm:

Algorithm 1 is the pseudo-code of our topic refinement algorithm. The algorithm takes as input a set **'Z'** of k topics each of n words i.e. $Z = \{z_1, ..., z_k\}$ where $z_i = \{w_{i_1}, ..., w_{i_n}\}$. Returns a set **'A'** of k aspects each of $n^{'}$ words i.e. $A = \{a_1, ..., a_k\}$ where $a_i = \{w_{i_1}, ..., w_{i_{n'}}\}$.

Algorithm 1 Coherent Cluster Growth (CCG) : Topics to aspects conversion

```
1: for $i \leftarrow 1, ..., k$ do
2:     $a_i \leftarrow (w_{i_x}, w_{i_y}) \leftarrow \text{bestPair}(z_i)$;
3:     for $j \leftarrow 1, ..., n^{'} - 2$ do
4:         $w_{i_u} \leftarrow \text{bestAddition}(a_i, z_i)$;
5:         $a_i \leftarrow a_i \cup w_{i_u}$;
6:         $z_i \leftarrow z_i - w_{i_u}$;
```

The algorithm starts by finding a pair of words that displays the highest semantic similarity in the given topic, see line 2. Different semantic similarity measures between words can be used. In this work we use cosine similarity of words as they appear in euclidean space.

Representation of words in vectorial form has proven success in practice in capturing semantic relations between words. Word embedding is a technique used in natural language processing where words or phrases are represented as vectors of real numbers [64–66]. Words close is semantic meaning tend to have similar vectors i.e. close points in euclidean space. We use this fact to refine extracted topics as described in algorithm 1.

After identifying the pair of words that displays the highest semantic similarity, pair members are set as initial elements of the aspect cluster a_i. For every iteration the algorithm then tries to grow the aspect cluster by finding a word within topic z_i which when added to the cluster will maximize the cluster coherence as computed by equation 3.1. $sim(w_i, w_j)$ is the similarity between two words.

$$ClusterCoherence = \frac{\sum_{i=2}^{n} \sum_{j=1}^{i-1} sim(w_i, w_j)}{\binom{n}{2}} \tag{3.1}$$

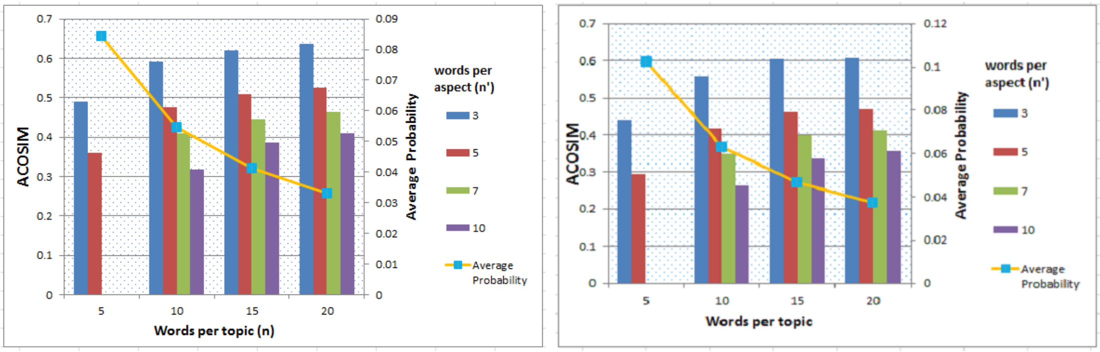
The best word found in line 4 is then appended to the aspect and removed from the topic. Iterations continue till a desirable number, $n^{'}$ of aspect representative words is reached. The procedure is then repeated to every topic in the topics set Z.

n and n' selection:

Algorithm 1 takes two hyper-parameters; number of words per topic, **n** and the desired number of words per aspect, **n'**. With larger values of **n** the risk of including words that actually do no belong to the topic increases. However, larger values of **n** increases the space from which coherent aspect-words are derived. Therefore, there is a constant trade-off between topic accuracy and aspect coherence when selecting the value of **n**.

Likewise too small value of **n'** faces the risk of concluding a wrong aspect from a given topic. An aspect should be emphasized by a reasonable number of words describing an identifiable item or concept. Larger value of **n'** however, reduces the aspect interpretability making an aspect nothing but a mere topic.

To regulate topic accuracy we use average probability of topic words. The idea is that, a topic should contain only the most probable words i.e. the average probability of topic-words should be high, this means smaller **n**. On the other hand we regulate the space for aspect extraction by using average pairwise cosine similarity (ACOSIM) of the extracted aspects. An aspect is required to have high ACOSIM, this can only be possible if **n** is high.



(a) hotels dataset

(b) phones dataset

Figure 3.2: number of words per topic against resulted ACOSIM

Figure 3.2 shows the hotels and cell phones dataset. For each dataset we experiment on different topic sizes, the number of words per topic, **n** is varied from 5, 10, 15 to 20. The number of topics, **k** is kept constant at 40 for both datasets as described in section 3.3. For each **n** we vary the demanded aspect size **n'** between 3,5,7 and 10 to determine the optimal aspect size, **n'**.

Experimental results in figure 3.2 show that, with the increase in **n**, ACOSIM increases for aspects of all sizes. However, the average probability of topic-words decreases. For n'=3 the two graphs intersection at n=7 and n'=3. For n'=5 the graphs intersect at n=10 and n'=5. To preserve aspect emphasis we optimize n and n' at n=10 and n'=5.

Topic refinement process:

To apply the topic refinement algorithm proposed in section 3.4.1, we therefore set n=10, n'= 5 and run the algorithm over topics returned by CPTM. We do not refine topics returned by ccLDA. This is because ccLDA topics contain both topic words (nouns) and words that show opinion about the topics (adjectives). This would require separation of nouns from other word-types before refinement, a task already performed by CPTM. For this reason we leave topics returned by ccLDA only to serve as baseline in evaluation.

Word vectors for similarity measurement are obtained from glove pre-trained embeddings[7] trained over common crawl (Google data). The embedding contain 2.2 Million words each of vector size 300.

Table 3.5 shows sample topics from cell phones dataset and their corresponding aspects after refinement.

Table 3.5: Sample topics and corresponding aspects

	Topic	Aspect
Topic22	anyone, m7, tell, photos, u, system, light, night, image, photo	photos, photo, image, light, night
Topic12	performance, s6, ram, core, lag, processor, cores, paper, cpu, games	core, cores, processor, cpu, ram
Topic8	apps, cant, music, download, app, feature, itunes, files, video, file	files, file, download, itunes, app
Topic37	quality, camera, features, pictures, sound, front, ones, speakers, cam, speaker	speakers, speaker, sound, quality, front
Topic13	camera, memory, resolution, size, mp, sensor, display, pixel, vs, iphone	resolution, pixel, display, sensor, camera

Note, in table 3.5 first five words in a topic are colored blue to indicate what an aspect would look like when top topic words are naively considered to be an aspect. The approach of considering top topic words as aspects is used in [13].

3.4.2 Aspects selection

Conversion of topics to aspects is usually not 100% successful. After refinement some aspects may remain unrecognizable i.e. some aspects may not show one coherent concept. Aspect selection aims at identifying aspects that are well refined. We define an aspect as well refined if it displays semantic coherence (average pairwise cosine similarity) greater or equal to 0.5.

[7] https://nlp.stanford.edu/projects/glove/

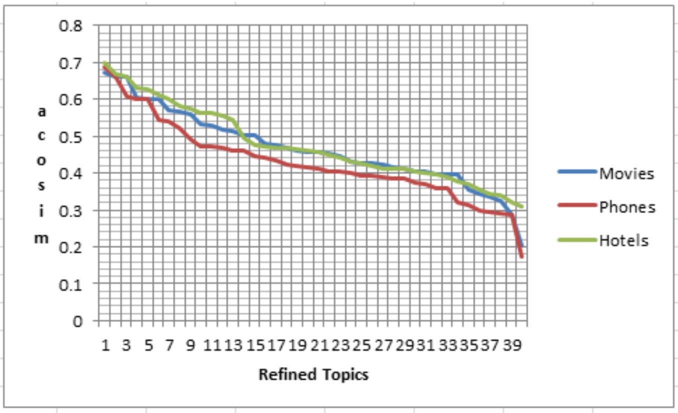


Figure 3.3: Topics sorted by their average pairwise cosine similarity (acosim)

Figure 3.4.2 shows topics from three datasets (phones, movies and hotels), sorted in decreasing order of their average pairwise cosine similarity. We can see that, for all datasets, the number of refined topics that qualify as aspects is between 10 to 15. This number conforms with expected number of aspects in the given datasets. For example, the hotels dataset has six predefined aspects i.e. reviewers comment on seven predefined aspects; value, room, location, cleanliness, check in/front desk, service and business service.

Interpretability of the selected aspects is remarkable. Resulted aspects from all datasets can be seen in appendix A. Figure 3.4 summarizes the stages taken to obtain final aspects from preprocessed corpus.

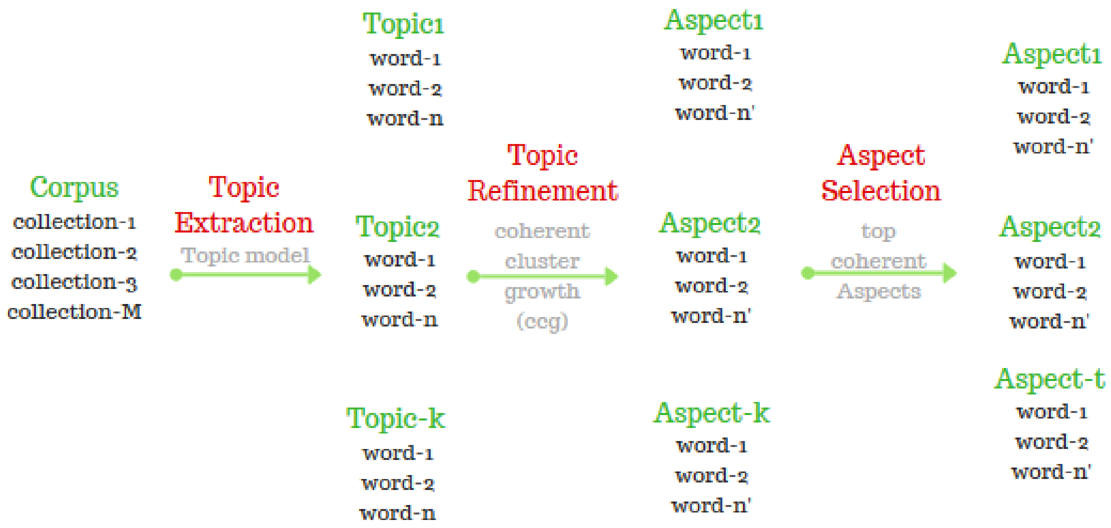


Figure 3.4: Aspects selection process

Figure 3.4 shows that, extraction of topics using topic models is not enough to regard topics as aspects, rather a refinement is performed to increase semantic intepretability of the topics. Moreover, not all refined topics are interpretable, therefore selection of best refined topics is done to obtain visually identifiable and semantic coherent aspects.

3.5 Sentiment classification

To determine sentiment orientation of the elicited aspects we feed opinion words associated to an aspect to the Valence Aware Dictionary and sEntiment Reasoner (VADER) [67]. VADER is a lexicon and rule based sentiment analysis tool implemented in python, java and php. To obtain opinion words associated to an aspect we use different methods depending on the core cross collection topic model in use.

When using CPTM, opinion words associated to an aspect are automatically mined as part of collection specific word distribution, we use this fact and select top 10 most probable words from this distribution for each topic.

For ccLDA returned collection specific words are a mixture of words that show opinions and words that don't. We therefore extract verbs, adverbs and adjectives in these distributions while ignoring nouns. This is because verbs, adverbs and adjectives tend of convey opinions, nouns on the other hand don't have this property.

Results of the top opinion words for each topic and each dataset for the two algorithms (ccLD and CPTM) is as illustrated in appendix A.

3.6 Results and Discussion

From results in appendix A; extracted aspects show that different algorithms give different weights to different topics in the same corpus. For example, in airlines dataset, a top topic according to ccLDA before refinement is topic 22, a topic on customer service, CPTM's top topic is about airplane fees and costs (topic12). When refinement is done, coherence of some topics that otherwise was low, increases. For example, in airlines a top aspect is aspect14, an aspect on baggage claiming. Selected aspects are highly identifiable and display latent structures that one may expect from the given corpora.

Chapter 4

Experimental Evaluation

In this chapter we evaluate the quality of aspect words and accuracy of sentiment scores returned by our proposed method. We start by performing intrinsic qualitative evaluation i.e. semantic interpretability of the aspects using pairwise mutual information (PMI) and cosine similarity, see section 2.4 for detailed information on these measures. We then evaluate the accuracy of sentiment scores associated to the returned aspects.

4.1 Aspect quality evaluation

To evaluate the quality of resulted aspects we use the held-out Wikipedia[1] corpus containing over a million articles. We conduct two experiments. First, we compare average pairwise cosine similarity of aspects returned by the CCG refinement algorithm to aspects returned by ccLDA and CPTM. Second we compare pairwise mutual information (PMI) of aspects returned by CCG refinement algorithm to those returned by ccLDA and CPTM. To achieve fair comparison the number of words representing an aspect in all of the three algorithms is set to 5.

[1] https://dumps.wikimedia.org/

4.1.1 Pairwise mutual information

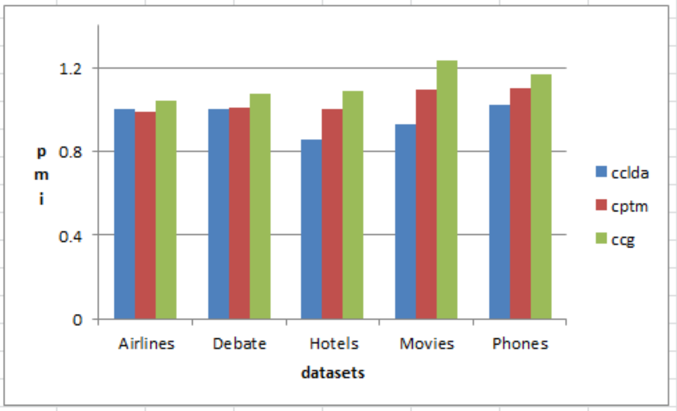


Figure 4.1: Average PMI scores for three algorithms on five datasets

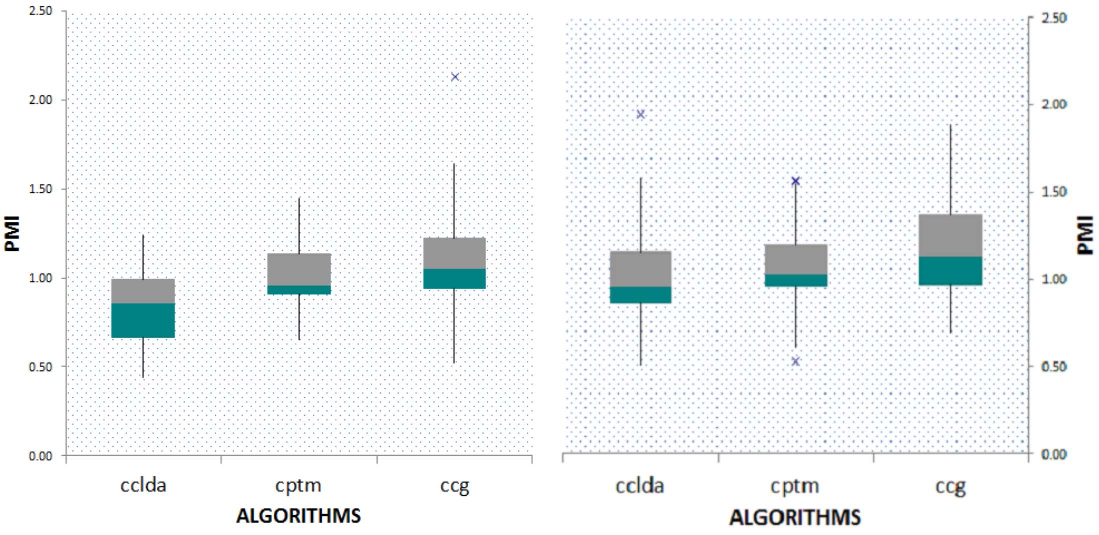
Figure 4.1 shows average pairwise mutual information (PMI) as measured over aspects returned by three aspect extraction algorithms i.e. ccLDA, CPTM and CCG. X-axis represents the datasets while y-axis represents PMI scores.

Note: datasets are arranged is increasing order of their sizes. As described in section 3.1, Airlines and debate fall under short documents, long collections. Hotels and movies have long documents but short collections while cell phones dataset is the largest dataset with long documents and large collections.

We can see that, with short documents large collections (airlines and dabate datasets), the interpretability of CPTM aspects is almost similar to that of ccLDA. This can be attributed to the fact that in short documents i.e. tweets, number of nouns when compared to number of all words in a corpus is not very different. For this reason, CPTM which bases its topic extraction on nouns will not have a significant improvement over ccLDA which bases its topic extraction on all word types. On the other hand, refinement of topics using CCG, offers a degree of improvement above CPTM and ccLDA. This improvement is however not very evident.

When considering short documents, large collections (hotels and movies datasets); the difference in interpetability of aspects extracted by ccLDA to those extracted by CPTM is amplified. In large collections recursive nouns are only a fraction of the entire vocabulary. Therefore, modeling nouns (CPTM) has a significant difference from modeling all word types. Moreover, with the wide option of nouns to model from, correct selection of those nouns as done by (CCG) offers a further improvement in aspect quality, the improvement is very evident and significant.

Finally, consider the PMI scores of cell phones dataset, the dataset consisting of long documents in huge collections. Due to the increase in number of documents, overall pmi score for both ccld and CPTM seems to have increased to match that of the first data category (airlines and debate). However, the increase in document length does not seem to impact interpretability. From this observation we can conclude that increase in document length is more impactful than increase in collection size.



(a) hotels dataset

(b) phones dataset

Figure 4.2: PMI box-plot for individual dataset

Figure 4.2 elaborates the variation of PMI scores within individual datasets. We can see that, when moving from left to right of both datasets; maximum, minimum and medium values improve. The median PMI score for CCG aspects is higher than those of ccLDA and CPTM. This means there are many identifiable aspects (aspects with PMI score above the median) from CCG than from ccLDA and CPTM.

4.1.2 Cosine similarity

Figure 4.3 shows average cosine similarity as measured over aspects returned by three aspect extraction algorithms i.e. ccLDA, CPTM and CCG. X-axis represents the datasets while y-axis represents average cosine similarity scores.

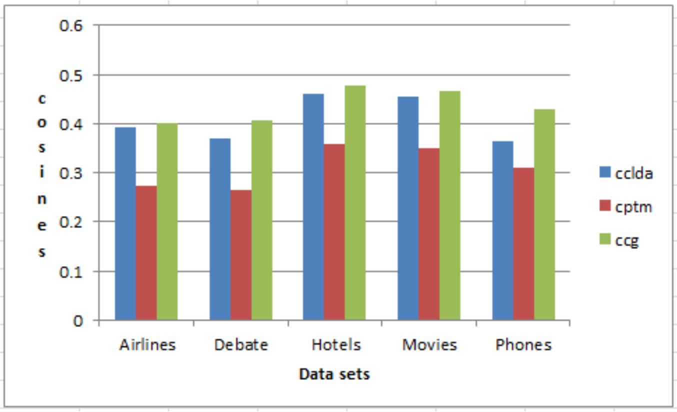
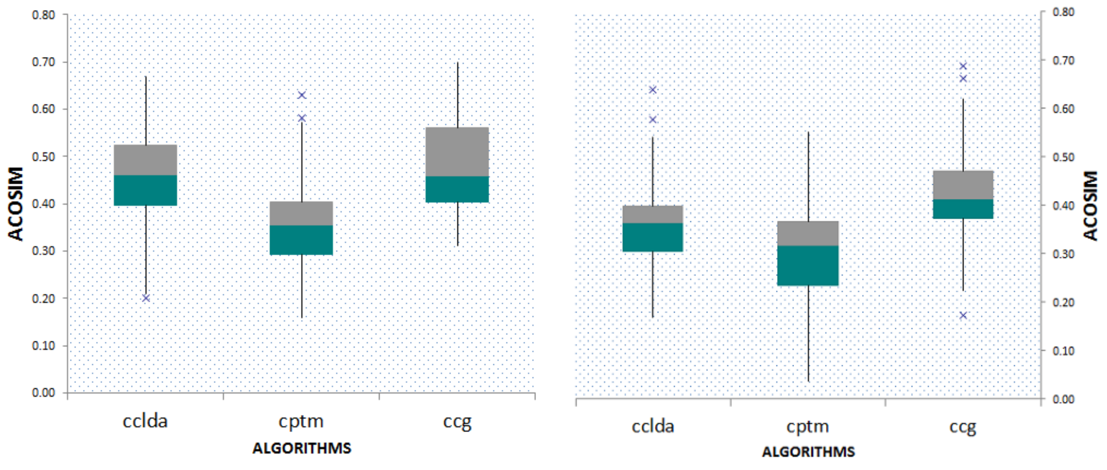


Figure 4.3: Average cosine similarity for three algorithms on five datasets

From figure 4.3 we see that, for all datasets, CCG outperforms ccLDA and CPTM in outputting semantic coherent aspects. The main difference between cosine scores and PMI scores presented in sub-section 4.1.1 is that, cosine similarity between two words considers number of times the two words occur in the same context in the given corpus, a context can be a sentence or a window of three to five words. On the other hand PMI considers number of documents the two words appear together in the given corpus.

This means cosine similarity represents a closer semantic interpretability than PMI. For example, a noun can be closer to a verb than how two nouns are closer to each other. Cosine similarity between *lunch* and *eat* (0.66) is higher than that between *lunch* and *chicken* (0.60). For this reason we have seen a significant drop in semantic coherence when modeling all words (ccLDA) from when modeling nouns only (CPTM). To counter this fall, enhancement of aspect intepretability using CCG is crusal.



(a) hotels dataset

(b) phones dataset

Figure 4.4: ACOSIM box-plot for individual datasets

Figure 4.4 elaborates the variation of ACOSIM scores within individual datasets. The figure shows that CPTM does not offer any improvement over ccLDA when ACOSIM is considered. The minimum, maximum and median ACOSIM scores of aspects drops when moving from ccLDA to CPTM. However, CCG improves over CPTM to match and even outperform ccLDA. Therefore, CCG not only considers nouns as aspects as opposed to all word types but also counters the interpretabilty distortion that comes from modeling nouns alone.

4.2 Sentiment scores evaluation

To evaluate the accuracy of sentiment scores assigned to elicited aspects we use the hotels dataset. Hotels dataset is preferred because it contains labeled aspect ratings for every document (review), i.e reviewers are asked to provide an aspect rating ranging from 1 to 5 on seven aspects; value, room, location, cleanliness, check in/front desk, service and business service for each review. These ratings serve as ground-truth for our evaluation.

Baseline sentiment scores:

We use the Aylien [2] implementation of [68] to compute baseline sentiment scores for each aspect. [68] proposed a hierarchical model of reviews for aspect based

[2] a rapidminer extension for aspect based opinion mining

sentiment analysis. The model is considered a state of the art as it has been used in multiple public applications including Aylien.

We execute the hierarchical model to three hotel collections one at a time. The hierarchical model identifies aspects and returns associated sentiment scores in a binary scale i.e. positive, negative or neutral for every aspect in every review. To obtain overall sentiment scores for each aspect we aggregate number of reviews that identify an aspect as positive, negative or neutral. A sentiment score is then given by equation 4.1, where ps, ng, nt is the number of reviews that identify an aspect **'a'** as positive, negative or neutral respectively.

$$sentiment(\mathbf{a}) = \frac{ps - ng}{ps + ng + nt} \tag{4.1}$$

Ground truth:

From the hotels dataset described in section 3.1 which contains labeled aspect ratings for every review, we aggregate and compute average over those ratings. Since the ratings range from 1 to 5 we map the final average to a value between -1 to 1 using equation $sentiment(x) = \frac{x}{2} - 1.5$, where x is a score between 1 and 5.

Table 4.1: Aspects and their representative words

Aspect	representative words
value	price, cost, money, value
rooms	bed, room, bathroom
location	location, street, city, distance
check in / front desk	staff, people, guests, reservation
services	drinks, food, buffet, lunch, restaurant

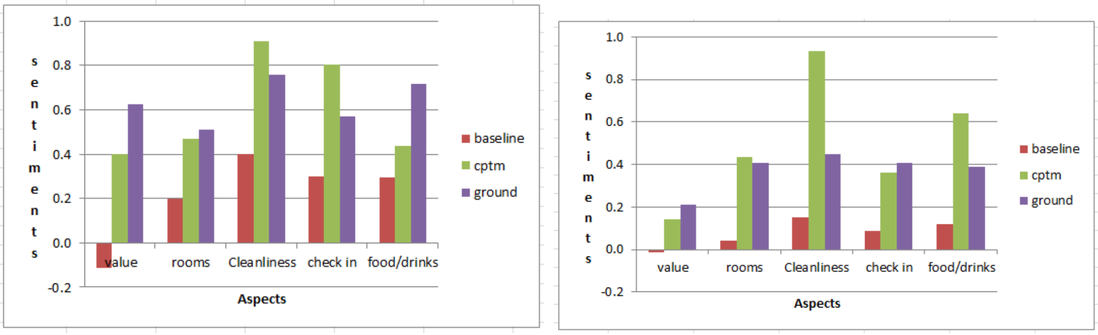
Table 4.1 shows aspects and their corresponding representative words. Representative words are words when seens in a topic, a topic can be said to represent a particular aspect.

Sentiment scores on five aspects as per ccLDA and CPTM with ground-truth ratings in parenthesis is as displayed in Table 4.2.

Table 4.2 and figure 4.5 shows that, for almost all aspects and hotels, sentiment scores computed using CPTM features outperform the baseline in matching the ground truth.

Table 4.2: Qualitative sentiment evaluation

Hotels	Aspects (baseline, CPTM, ground truth)				
	value	rooms	cleanliness	check in	food/drinks
Caribe club	-0.09, 0.66, (0.39)	0.01, 0.44, (0.28)	0.28, 0.46, (0.42)	0.14, -0.13, (0.40)	0.24, 0.30, (0.39)
Iberostar	-0.11, 0.40, (0.62)	0.20, 0.47, (0.51)	0.40, 0.91, (0.76)	0.30, 0.80, (0.57)	0.30, 0.44, (0.72)
Rio Mar	-0.01, 0.14, (0.21)	0.04, 0.43, (0.41)	0.15, 0.93, (0.45)	0.09, 0.36, (0.41)	0.12, 0.64, (0.39)



(a) sentiment scores for ibero star

(b) sentiment scores for riomar

Figure 4.5: Average cosine similarity for long documents, small collection

Root Mean Square Error (RMSE) of CPTM features versus baseline scores as measured against the ground truth is as summarized in figure 4.6.

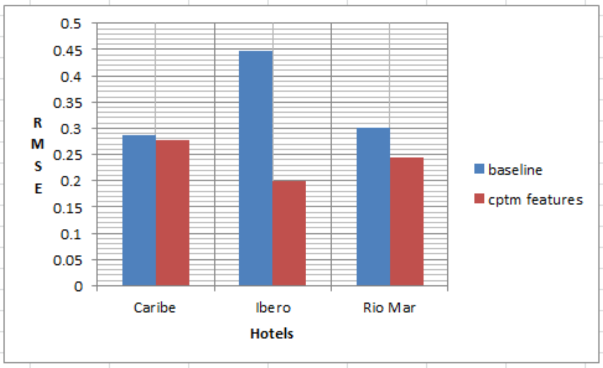


Figure 4.6: RMSE of sentiment scores of baseline versus CPTM features

With this observation we can conclude that, CPTM features when used with sentiment lexicon, yields more accurate sentiment score than the hierrachical model.

4.3 Results and Discussion

In this section we evaluated the quality of aspects extracted from our proposed aspect extraction method. We further evaluated the accuracy of the sentiment scores assigned to those aspects.

Results indicated that, The root mean square error (RMSE) of sentiment scores measured from CPTM features against the ground truth is significantly lower than the RMSE between the baseline and the ground truth. This has been true for all three hotels presented for evaluation.

On the other hand, results on aspect extraction show that, aspects from proposed topic refinement algorithm (CCG) show remarkable higher semantic coherence than ccLDA and CPTM topics.

Chapter 5

Conclusion and Future Work

This work combined sentiment analysis and cross collection topic modeling to form a framework that performs aspect based sentiment analysis simultaneously across comparable document collections.

The task is presented as a two phase problem, first is the extraction of aspects that prevail across multiple collections and second is the association of those aspects to their numerical sentiment scores.

For aspect extraction we used topic models, ccLDA and CPTM to extract topics that prevail across all desirable collections. We then presented a topic refinement algorithm named coherent cluster growth (CCG) that successfully converts the extracted topics into visually identifiable aspects. Results show that CCG aspects show remarkable higher semantic coherence than ccLDA and CPTM topics.

For sentiment orientation of elicited aspects we performed lexicon based sentiment analysis using opinion words returned by CPTM as features. Results show that these features resulted to better sentiment accuracy than the hierarchical model.

A possible future work may include integration of semantic consideration within the structure of the cross collection topic models. This means making topic refinement a process with topic extraction instead of considering topic refinement as a post processing process.

Finally, since it is an expensive task to annotate/label polarity of each aspect and their corresponding entities, most of the available labeled datasets for aspect based sentiment analysis are small. It is almost impossible to test the performance of algorithms to large datasets (they do not exist). Therefore, development of evaluation measures that do not require labeled data is a noble course.

Appendix A

Output of the three algorithms on each of the datasets

Results from the Airlines dataset

Table A.1: ccLDA results for airlines dataset

	Topic words	Perspectives words				
		United	American	Delta	USAirways	Southwest
Topic22	service, customer, rude, terrible, rep	excited incredible protocol fool duty	conflicting pathetic follows hooked operational	redeemed guilty united visible available	disorganization prerecorded gloves fearing insult	sympathy deadhead waiting processes bunch
Topic25	seat, first, seats, class, available	big squished delinquent picked rest	portable smoothest alive fleet reconsider selected	hanger booze slowly regulations economy	consistent unserviceable attention guess reunion	unnoticed hanger unaccompanied pathetic refunding
Topic28	yesterday like bags luggage baggage	madness, pig institutional gorgeous	heavily awesome fewer comparable empty	appropriate, overbooked, brilliant	hardest liar crowd fight	casual, gassing, differently issuing destroying
Topic12	flight, late, hours, seats, delayed	headaches rely accountability	preregistration scared unused	fails apology constantly approach	manners frustrations erroring sounds	useful help intentionally forcing
Topic2	email, number, doesnt, info, website	locked appropriate linking strong dark	blew planned chaotic crashed invalid	active intended directtv reference acted	danger special unclear prime force	incredibly significant superiors

Table A.2: CPTM results for airlines dataset

	Topic words	Perspectives words				
		United	American	Delta	USAirways	Southwest
Topic12	change, ticket, fee, car, request	cancelled booked flighted	cant even possible	cant really frustrating	trying even got	quick faster able
Topic14	hour, baggage, bags, claim, issue,	delayed lost waiting	missing long sitting	lost waiting new	still lost cancelled	checked delivered going
Topic8	phone, number, reservation, confirmation, someone	rebooked flightled	cancelled worked	really horrible	called booked boarded	cancelled booked
Topic30	customer, service, information, relations, advice	terrible poor worst disappointing	terrible worst ever	great amazing excellent	worst terrible horrible	great terrible disappointing amazing
Topic37	flight, attendant, crew, pilot, board	cancelled late delayed	cancelled flightled flighted	great working extra	late unacceptable delayed	great boarding free

Table A.3: CCG results for airlines dataset

	Aspect words	Perspectives words				
		United	American	Delta	USAirways	Southwest
Aspect14	baggage, bags, claim, issue, wait	delayed lost waiting	missing long sitting	lost waiting new	still lost cancelled	checked delivered going
Aspect12	ticket, fee, price, system, change	cancelled booked flighted	cant even possible	cant really frustrating	trying even got	quick faster able
Aspect21	agent, agents, people, phone, desk	delayed long waiting	horrible late wait	delayed estimated listed	scheduled connecting boarded	first still last long
Aspect30	service, customer, information, relations, advice	terrible poor worst disappointing	terrible worst ever	great amazing excellent	worst terrible horrible	great terrible disappointing amazing
Aspect37	flight, crew, pilot, attendant, board	cancelled late delayed	cancelled flightled flighted	great working extra	late unacceptable delayed	great boarding free

Results from the Debate dataset

Table A.4: ccLDA results for debate dataset

	Topic words	Perspectives words	
		Hillary	Donald
Topic1	question, policy, foreign, speech, convention	pessimistic, generous	totally, biased
Topic3	women, men, fair, pay, jobs, economy	powerful, strong, hoped	strong, announced, live
Topic11	would, trumps, tax, plan, returns	richest homes benefits enough	increase refuses allow lie
Topic19	need stop gun violence guns	strong, preventing, reduce	immediately, replace, hispanic
Topic36	question wall immigration build immigrants	undocumented, add, financial	illegal, easily, discussing

Table A.5: CPTM results for debate dataset

Topics	Topic words	Hillary Clinton	Donald Trump
Topic3	trump tax isis return donald	fair, give, american	never, ever, said
Topic8	women job business proud time	unfit, good, american	amazing, good, crooked
Topic12	job, work ,pay, share	together, equal	new, crooked
Topic20	everyone, law, police, officers, enforcement	equal, peaceful	amazing, fantastic, many
Topic39	care, plan, family, dept, college	paid, affordable, free	new, amazing, better

Table A.6: CCG results for debate dataset

Aspect	Aspect words	Hillary Clinton	Donald Trump
Aspect3	tax, taxes, return, share, judge	fair, give, american	never, ever, said
Aspect4	justice, act, fight, crisis, movement	lost, better, working, together	really, allowed, happy, wrong
Aspect10	pennsylvania, wisconsin ,immigration, polls, rally	hateful, progressive	good, big, never, soon
Aspect20	police, officers, enforcement, law, order	equal, peaceful	amazing, fantastic, many
Aspect39	care, health, plan, child, family	paid, affordable, free	new, amazing, better

Results from the Movies dataset

Table A.7: ccLDA results for movies dataset

Topics	Topic words	Apocalypto	Gran torino	The notebook	Schoolhouse	Alexander
Topic1	great, scenes, much, good, acting	conservative conflicted overrated	highly recommend thrilled	beheadings religious thrilling	confusing disaster weird	anxiously teenageer kissing
Topic25	dvd, ray, original, blu	verbally board worked virtually	nice happy generation songs	absolute, authentic	paced, theatrical, revisited	recommending, wanted, agreed
Topic26	story actors characters well acting	grizzled imagined humorous	appropriate memorized compilation	visually unique realistic exotic	inaccuracies dramatically obsession	gosling classic imagination
Topic12	love story old girl home	persistent morality	remembered engaging	pursuers devoted	biopic approach	parents wealthy allies
Topic31	director production best picture hollywood	changeling	annoying, bombarded	indigeneous, detailed, overwhelmed	total, sadly, narrative	financial

Table A.8: CPTM results for movies dataset

Topics	Topic words	Apocalypto	Gran torino	The notebook	Schoolhouse	Alexander
Topic14	role, actor, performance, actors, screen	best, good, fine	perfect, great, truly	well, beautiful, natural	quationable, banned, confusing	bad, worst, utterly
Topic34	version, ray, blu, quality, anyone	good amazing best	really good excellent	absolutely, amazing, beautiful	incredible, good, enjoying	great, good, high
Topic18	scenes, scene, battle, directors, cut	good, chase, watching, final	really, unfamiliar, good, memorable	deleted, suddenly, rated, pg	simplistic, young, daughter	theatrical, historical, really, original
Topic8	audience, questions, sex, men, women	immediately, showing, simple	cultural, decent, asian	almost, often, feeling	good, basically, together	real, bisexual, sexual,
Topic38	movie, movies, watch, trailers, research	good, great, seen	enjoyed, excellednt, long	romantic, great, good	disappointed, regular, classic	good, great, best,

Table A.9: CCG results for movies dataset

Aspects	Aspect words	Apocalypto	Gran torino	The notebook	Schoolhouse	Alexander
Aspect1	civilization, culture, violence, passion, christ	well, violent, great	nice, modern, educates	starred, cherished, emotionally	enthralled, mesmerized, best	insult, implying, leading
Aspect12	song, songs, rock, kids, collection	naked, lucky, pretty	disappointed, changeling, worst	heartbreaking, definitely, worth	great, learned, lolly	ill, close, succeeded
Aspect18	scenes, scene, battle, directors, cut	good, chase, watching, final	really, unfamiliar, good, memorable	deleted, suddenly, rated, pg	simplistic, young, daughter	theatrical, historical, really, original
Aspect14	role, roles, actors, actor, character	excellent, best, gifted	powerful, perfect, great	beautiful, rich, natural	excellent, amazing, questionable	worst, utterly, bad,
Aspect38	movie, movies, watch, trailers, research	good, great, seen	enjoyed, excellent, long	romantic, great, good	disappointed, regular, classic	good, great, best,

Results from the Hotels dataset

Table A.10: ccLDA results for hotels dataset

Topics	Topic words	Iberostar	Rio mar	caribe club
Topic7	food, restaurants, restaurant, good, buffet	attractive, incredibly, upscale	liquor scratch frustrating	diarrhoea insulting losing
Topic11	trip went day island excursion	rainforest hiking attractions fajardo	greatest massage advantage	bumpy hockey horse
Topic17	room ocean bed beds two	large secluded heavenly	awful unfortunately mattress	honor entertain excess
Topic19	room front desk service door	awaiting, witnessed	certainly, upscale	casual, designed
Topic28	beach, water, sand, lots, white	cool brown nice	purified discourage bothering	humidity ponds shorts

Table A.11: CPTM results for hotels dataset

Aspects	Aspect words	caribe club	Iberostar	Rio mar
Topic4	person, prices, price, minutes, pay	inclusive, worth, wonderful, better, ask	expensive, less, allowed, close, funny	free, nice, little, high, take, away
Topic15	room, bed, beds, door, check	amazingly, clean, favorite	really, quiet, double, king, booked	king, called, told, never, given
Topic8	resort, golf, course, resorts, courses	beautiful, friendly, wonderful, good	beautiful, wonderful, amazing, used	excellent, amazing, beautiful, great
Topic26	room, night, dinner, breakfast, nights	definitely, good, great, available	spanish, everyday, good, beautiful, close	best, clean, first, still, given
Topic13	trip, time, island, fun, boat	little, fun, great, well, last	great, always, entertaining, much, snorkeling	great, good, loved, rainforest, snorkeling

Table A.12: CCG results for hotels dataset

Aspects	Aspect words	caribe club	Iberostar	Rio mar
Aspect4	prices, price, cost, pay, deal	inclusive, worth, wonderful, better, ask	expensive, less, allowed, close, funny	free, nice, little, high, take, away
Aspect15	bed, beds, room, door, reception	amazingly, clean, favorite	really, quiet, double, king, booked	king, called, told, never, given
Aspect26	dinner, lunch, breakfast, buffet, chicken	definitely, good, great, available	spanish, everyday, good, beautiful, close	best, clean, first, still, given
Aspect25	hotel, facilities, pool, beach, restaurant	great, much, better, overall, inclusive	good, great, much, fine, different	great, good, nice, enough, friendly
Aspect13	excursion, excursions, trips, trip, boat	little, fun, great, well, last	great, always, entertaining, much, snorkeling	great, good, loved, rainforest, snorkeling

Results from the Phones dataset

Table A.13: ccLDA results for phones dataset

Topics	Topic words	Blackberry	Galaxy	HTC one M7	Iphone6	Sony xperia
Topic13	screen phone big size bigger	old, small, terrible		full, better	follow	clear, flat, easy
Topic25	problem phone using problems issue	battery heating issues		software working problem		missing electronics heating electronics
Topic20	camera take front better quality	good, nice,clear		common, fixed, related		better low
Topic21	phone gb 64gb enough memory	done, downloaded	removable, expandable		fixed, everywhere	
Topic38	ram core performance cores game	single, unable		much, less		enough, realy, nice

Table A.14: CPTM results for phones dataset

Topics	Topic words	Iphone6	HTC one M7	Blackberry	Galaxy	Sony xperia
Topic8	apps, music, download, app, feature	free, downloading	best, apps, updated	many, apps, cant, downloaded	single, well, longer	free, downloading, old
Topic10	issue, services, network, customer, center	best, fixed, replace	replaced, new, best	drained, replace, last	overheated, exploading, replace	poor, useless, freezing
Topic12	performance, s6, ram, core, lag	fast, better, dual	good, better, faster	definitely, brilliant, excellent	good, well, faster	single, fast, better
Topic39	phones, smartphone, market, share, purchase	best, better, actually, good	best, good, better, high	best, android, ever	best, samsung, mobile, good	best, nexus, great, selling
Topic37	quality, camera, features, pictures, sound	good, great, best	good, great, amazing	good, great, better	better, great, less	bad, good, nice

Table A.15: CCG results for phones dataset

Aspects	Aspect words	Iphone6	HTC one M7	Blackberry	Galaxy	Sony xperia
Aspect8	files, file, download, itunes, app	free, downloading	best, apps, updated	many, apps, cant, downloaded	single, well, longer	free, downloading, old
Aspect10	services, customer, network, repair, warranty	best, fixed, replace	replaced, new, best	drained, replace, last	overheated, exploading, replace	poor, useless, freezing
Aspect12	core, cores, processor, cpu, ram	fast, better, dual	good, better, faster	definitely, brilliant, excellent	good, well, faster	single, fast, better
Aspect13	resolution, pixel, display, sensor, camera	great, perfect, best	bad, fix, low	poor, secondary, good	better, great, best	higher, clear, perfect
Aspect37	speakers, speaker, sound, quality, front	good, better, best	amazing, really, awesome	great, quite, fine	great, less, good	bad, good, better

Bibliography

[1] B. Liu and L. Zhang, "A survey of opinion mining and sentiment analysis," in *Mining text data.* Springer, 2012, pp. 415–463.

[2] S. Brody and N. 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.* Association for Computational Linguistics, 2010, pp. 804–812.

[3] Y. Jo and A. 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.* ACM, 2011, pp. 815–824.

[4] C. Lin and Y. He, "Joint sentiment/topic model for sentiment analysis," in *Proceedings of the 18th ACM conference on Information and knowledge management.* ACM, 2009, pp. 375–384.

[5] Y. Liu, X. Huang, A. An, and X. Yu, "Arsa: a sentiment-aware model for predicting sales performance using blogs," in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 2007, pp. 607–614.

[6] Y. Lu and C. Zhai, "Opinion integration through semi-supervised topic modeling," in *Proceedings of the 17th international conference on World Wide Web.* ACM, 2008, pp. 121–130.

[7] Y. Lu, C. Zhai, and N. Sundaresan, "Rated aspect summarization of short comments," in *Proceedings of the 18th international conference on World wide web.* ACM, 2009, pp. 131–140.

[8] Q. Mei, X. Ling, M. Wondra, H. Su, and C. Zhai, "Topic sentiment mixture: modeling facets and opinions in weblogs," in *Proceedings of the 16th international conference on World Wide Web.* ACM, 2007, pp. 171–180.

[9] I. Titov and R. T. McDonald, "A joint model of text and aspect ratings for sentiment summarization." in *ACL*, vol. 8, 2008, pp. 308–316.

[10] I. Titov and R. McDonald, "Modeling online reviews with multi-grain topic models," in *Proceedings of the 17th international conference on World Wide Web.* ACM, 2008, pp. 111–120.

[11] H. Wang, Y. Lu, and C. 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.* ACM, 2010, pp. 783–792.

[12] W. X. Zhao, J. Jiang, H. Yan, and X. Li, "Jointly modeling aspects and opinions with a maxent-lda hybrid," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing.* Association for Computational Linguistics, 2010, pp. 56–65.

[13] N. Naveed, T. Gottron, and S. Staab, "Feature sentiment diversification of user generated reviews: The freud approach," in *Seventh International AAAI Conference on Weblogs and Social Media*, 2013.

[14] D. Freitag and A. McCallum, "Information extraction with hmm structures learned by stochastic optimization," *AAAI/IAAI*, vol. 2000, pp. 584–589, 2000.

[15] W. Jin and H. H. H. A. N. Lexicalized, "Hmm-based learning framework for web opinion mining in proceedings of the 26th international conference on machine learning," *Montreal, Canada*, 2009.

[16] W. Jin, H. H. Ho, and R. K. Srihari, "Opinionminer: a novel machine learning system for web opinion mining and extraction," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 2009, pp. 1195–1204.

[17] N. Jakob and I. Gurevych, "Extracting opinion targets in a single-and cross-domain setting with conditional random fields," in *Proceedings of the 2010 conference on empirical methods in natural language processing.* Association for Computational Linguistics, 2010, pp. 1035–1045.

[18] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[19] M. Hu and B. L. Mining, "Summarizing customer reviews kdd 04, august 22–25, 2004," *Seattle, Washington, USA.*

[20] B. Liu, M. Hu, and J. Cheng, "Opinion observer: analyzing and comparing opinions on the web," in *Proceedings of the 14th international conference on World Wide Web.* ACM, 2005, pp. 342–351.

[21] M. Paul and R. Girju, "Cross-cultural analysis of blogs and forums with mixed-collection topic models," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3.* Association for Computational Linguistics, 2009, pp. 1408–1417.

[22] Y. Fang, L. Si, N. Somasundaram, and Z. Yu, "Mining contrastive opinions on political texts using cross-perspective topic model," in *Proceedings of the fifth ACM international conference on Web search and data mining.* ACM, 2012, pp. 63–72.

[23] C. Zhai, A. Velivelli, and B. Yu, "A cross-collection mixture model for comparative text mining," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 2004, pp. 743–748.

[24] S. T. Dumais, "Latent semantic indexing (lsi): Trec-3 report," *Nist Special Publication SP*, pp. 219–219, 1995.

[25] T. Hofmann, "Probabilistic latent semantic analysis," in *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence.* Morgan Kaufmann Publishers Inc., 1999, pp. 289–296.

[26] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[27] W. Li and A. McCallum, “Pachinko allocation: Dag-structured mixture models of topic correlations,” in *Proceedings of the 23rd international conference on Machine learning.* ACM, 2006, pp. 577–584.

[28] L. Fei-Fei and P. Perona, “A bayesian hierarchical model for learning natural scene categories,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 524–531.

[29] S. Rogers, M. Girolami, C. Campbell, and R. Breitling, “The latent process decomposition of cdna microarray data sets,” *IEEE/ACM transactions on computational biology and bioinformatics*, vol. 2, no. 2, pp. 143–156, 2005.

[30] T. Masada, T. Hamada, Y. Shibata, and K. Oguri, “Bayesian multi-topic microarray analysis with hyperparameter reestimation,” *Advanced Data Mining and Applications*, pp. 253–264, 2009.

[31] L. P. Coelho, T. Peng, and R. F. Murphy, “Quantifying the distribution of probes between subcellular locations using unsupervised pattern unmixing,” *Bioinformatics*, vol. 26, no. 12, pp. i7–i12, 2010.

[32] M. Bicego, P. Lovato, A. Ferrarini, and M. Delledonne, “Biclustering of expression microarray data with topic models,” in *Pattern Recognition (ICPR), 2010 20th International Conference on.* IEEE, 2010, pp. 2728–2731.

[33] N. Pratanwanich and P. Lio, “Exploring the complexity of pathway–drug relationships using latent dirichlet allocation,” *Computational biology and chemistry*, vol. 53, pp. 144–152, 2014.

[34] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, “Indexing by latent semantic analysis,” *Journal of the American society for information science*, vol. 41, no. 6, p. 391, 1990.

[35] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, “Text classification from labeled and unlabeled documents using em,” *Machine learning*, vol. 39, no. 2, pp. 103–134, 2000.

[36] H. M. Wallach, “Topic modeling: beyond bag-of-words,” in *Proceedings of the 23rd international conference on Machine learning.* ACM, 2006, pp. 977–984.

[37] D. M. Blei and J. D. Lafferty, "A correlated topic model of science," *The Annals of Applied Statistics*, pp. 17–35, 2007.

[38] T. L. Griffiths, M. I. Jordan, J. B. Tenenbaum, and D. M. Blei, "Hierarchical topic models and the nested chinese restaurant process," in *Advances in neural information processing systems*, 2004, pp. 17–24.

[39] J. D. Mcauliffe and D. M. Blei, "Supervised topic models," in *Advances in neural information processing systems*, 2008, pp. 121–128.

[40] S. Lacoste-Julien, F. Sha, and M. I. Jordan, "Disclda: Discriminative learning for dimensionality reduction and classification," in *Advances in neural information processing systems*, 2009, pp. 897–904.

[41] J. Zhu, A. Ahmed, and E. P. Xing, "Medlda: maximum margin supervised topic models," *Journal of Machine Learning Research*, vol. 13, no. Aug, pp. 2237–2278, 2012.

[42] D. Ramage, D. Hall, R. Nallapati, and C. 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.* Association for Computational Linguistics, 2009, pp. 248–256.

[43] D. Ramage, C. D. Manning, and S. Dumais, "Partially labeled topic models for interpretable text mining," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 2011, pp. 457–465.

[44] Y. Petinot, K. McKeown, and K. Thadani, "A hierarchical model of web summaries," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2.* Association for Computational Linguistics, 2011, pp. 670–675.

[45] V.-A. Nguyen, J. L. Boyd-Graber, and P. Resnik, "Lexical and hierarchical topic regression," in *Advances in neural information processing systems*, 2013, pp. 1106–1114.

[46] S. S. Kataria, K. S. Kumar, R. R. Rastogi, P. Sen, and S. H. Sengamedu, "Entity disambiguation with hierarchical topic models," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 2011, pp. 1037–1045.

[47] A. Bakalov, A. McCallum, H. Wallach, and D. Mimno, "Topic models for taxonomies," in *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries.* ACM, 2012, pp. 237–240.

[48] X.-L. Mao, Z.-Y. Ming, T.-S. Chua, S. Li, H. Yan, and X. Li, "Sshlda: a semi-supervised hierarchical topic model," in *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning.* Association for Computational Linguistics, 2012, pp. 800–809.

[49] M. Paul and R. Girju, "A two-dimensional topic-aspect model for discovering multi-faceted topics," *Urbana*, vol. 51, no. 61801, p. 36, 2010.

[50] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10.* Association for Computational Linguistics, 2002, pp. 79–86.

[51] L. Qiu, W. Zhang, C. Hu, and K. Zhao, "Selc: a self-supervised model for sentiment classification," in *Proceedings of the 18th ACM conference on Information and knowledge management.* ACM, 2009, pp. 929–936.

[52] M. Eirinaki, S. Pisal, and J. Singh, "Feature-based opinion mining and ranking," *Journal of Computer and System Sciences*, vol. 78, no. 4, pp. 1175–1184, 2012.

[53] C. Lin, Y. He, R. Everson, and S. Ruger, "Weakly supervised joint sentiment-topic detection from text," *IEEE Transactions on Knowledge and Data engineering*, vol. 24, no. 6, pp. 1134–1145, 2012.

[54] B. Pang and L. Lee, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," in *Proceedings of the 43rd*

annual meeting on association for computational linguistics. Association for Computational Linguistics, 2005, pp. 115–124.

[55] S. Ahire, "A survey of sentiment lexicons," 2014.

[56] M. M. Bradley and P. J. Lang, "Affective norms for english words (anew): Instruction manual and affective ratings," Citeseer, Tech. Rep., 1999.

[57] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[58] C. J. Wu, "On the convergence properties of the em algorithm," *The Annals of statistics*, pp. 95–103, 1983.

[59] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proceedings of the National academy of Sciences*, vol. 101, no. suppl 1, pp. 5228–5235, 2004.

[60] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-Graber, and D. M. Blei, "Reading tea leaves: How humans interpret topic models," in *Advances in neural information processing systems*, 2009, pp. 288–296.

[61] D. Newman, S. Karimi, and L. Cavedon, "External evaluation of topic models," in *in Australasian Doc. Comp. Symp., 2009.* Citeseer, 2009.

[62] G. Bouma, "Normalized (pointwise) mutual information in collocation extraction," *Proceedings of GSCL*, pp. 31–40, 2009.

[63] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. McCallum, "Optimizing semantic coherence in topic models," in *Proceedings of the conference on empirical methods in natural language processing.* Association for Computational Linguistics, 2011, pp. 262–272.

[64] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of machine learning research*, vol. 3, no. Feb, pp. 1137–1155, 2003.

[65] J. Pennington, R. Socher, and C. Manning, “Glove: Global vectors for word representation,” in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[66] O. Levy and Y. Goldberg, “Dependency-based word embeddings,” in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, vol. 2, 2014, pp. 302–308.

[67] C. H. E. Gilbert, “Vader: A parsimonious rule-based model for sentiment analysis of social media text,” in *Eighth International Conference on Weblogs and Social Media (ICWSM-14). Available at (20/04/16) http://comp. social. gatech. edu/papers/icwsm14. vader. hutto. pdf*, 2014.

[68] S. Ruder, P. Ghaffari, and J. G. Breslin, “A hierarchical model of reviews for aspect-based sentiment analysis,” *arXiv preprint arXiv:1609.02745*, 2016.