Hidden Topic Sentiment Model

Md Mustafizur Rahman, Hongning Wang
Department of Computer Science
University of Virginia
Charlottesville VA, 22903 USA
{mr4xb, hw5x}@cs.virginia.edu

ABSTRACT

Various topic models have been developed for sentiment analysis tasks. But the simple topic-sentiment mixture assumption prohibits them from finding fine-grained dependency between topical aspects and sentiments. In this paper, we build a Hidden Topic Sentiment Model (HTSM) to explicitly capture topic coherence and sentiment consistency in an opinionated text document to accurately extract latent aspects and corresponding sentiment polarities. In HTSM, 1) topic coherence is achieved by enforcing words in the same sentence to share the same topic assignment and modeling topic transition between successive sentences; 2) sentiment consistency is imposed by constraining topic transitions via tracking sentiment changes; and 3) both topic transition and sentiment transition are guided by a parameterized logistic function based on the linguistic signals directly observable in a document. Extensive experiments on four categories of product reviews from both Amazon and NewEgg validate the effectiveness of the proposed model.

General Terms

Algorithm, Experimentation

Keywords

Topic modeling, aspect detection, sentiment analysis

1. INTRODUCTION

Topic models have become an important building block in sentiment analysis [18, 12, 11, 15, 31, 28]. It naturally decomposes unstructured text content into topical aspects and sentiment polarities via generative modeling. The automatically identified topics and corresponding opinions provide a fine-grained understanding of opinionated text data and enable a wide range of important applications, including public opinion tracking in social media [18, 15, 12], automated recommendation in e-commerce [16], contrastive opinions summarization in political science [6], and many more.

One fundamental assumption in topic models is *exchangeability*, i.e., topics are infinitely exchangeable within a given document while the joint probability is invariant [3]. As a result, a common practice is to model a document as a mixture over a set of latent topics; and given topic mixing proportion, the topic assignments over words in a document are considered as independent from each other. This overly simplified assumption fails to capture rich structures embedded in a text document: in reality, natural language text rarely consists of isolated, unrelated sentences, but rather collocated, structured and coherent groups of sentences [10]. The existence of sentiment in an opinionated text document further increases the complication of topic and sentiment mixture. For example, most topic models for sentiment analysis assume the selection of topics are independent given sentiment labels over words [18, 15, 12]. However, it is very unlikely for a user to express contradictory sentiment, i.e., both positive and negative, about the same topical aspect in a document; and thus when sentiment switches, the topic should also change. Enhanced independence assumption is expected to yield better models in terms of latent aspect identification and sentiment classification. Figure 1 illustrates this interdependency between topic assignments and sentiment polarities in a typical product review, which motivates our research in this paper.

By: Kindle Customer **Date:**June 25, 2014
I own an ultrabook and I like it for a number of specific tasks. I especially like its portability (3 pounds with a small footprint) {portability,+} and the speed of its solid state drive {hard drive,+}. When it comes to looks you have to give it to the Inspiron {appearance,+}. It definitely has the sleek look of an ultrabook {appearance,+}. The combination of brushed aluminum with black trim, keys and bezel make for a very classy, "corporate" presence {appearance,+}. The fit and finish are first rate {appearance,+}. However, the sound sucks{sound,-}. I have owned 10 notebook and laptop computers over the past two decades and this Inspiron has the worst sound of any before it {sound,-}. It is weak, tinny and what low end it has is muddy and indistinct {sound,-}. While we've all come to expect pretty lousy sound from notebooks, this is subpar even considering those low standards {sound,-}.

Figure 1: A review of a laptop from Amazon[1]. Topical aspects and sentiment polarities are manually labeled in superscripts with different colors on each sentence.

Three important observations can be found in the sample review document annotated in Figure 1. First, topic assignments over words in a document are not a simple mixture; instead, words in close proximity tend to share the same topic, i.e., *topic coherence*. Second, sentiment polarities expressed toward the same topical aspect tend to be consistent, i.e., *sentiment consistency*. We should note that this observation is not contradicting with the fact that a user might have mixed judgements about an item within a review

[1] http://www.amazon.com/gp/customer-reviews/RQ4YYC5BXD021

WWW 2016, April 11–15, 2016, Montréal, Québec, Canada.
ACM 978-1-4503-4143-1/16/04.
http://dx.doi.org/10.1145/2872427.2883072 .

document, e.g., appreciate appearance but dislike sound quality of the ultrabook in our motivating example. Sentiment consistency suggests that a user tends to give the same opinion about a particular topical aspect, rather than expressing contradictory assessments over it. This adds another dimension of regularity of topic assignments over words in an opinionated text document: when sentiment switches, the topic assignment should also switch. Last but not least, there are clear linguistic cues indicating the transition of sentiment and topics between successive sentences. For example, conjunctions like "however" and "nonetheless" imply a switch of sentiment in the current sentence, while an increased overlap of content words suggest unaltered topic and sentiment assignment between two adjacent sentences.

Some solutions have been developed to realize topic coherence, i.e., assign words in a sentence to the same topic [12] and model topic transition among successive sentences [8, 29]. Linguistic cues, e.g., POS tagging [31] and metadata [19], have been also exploited to guide topic generation. But exchangeability assumption is still being made when modeling the compound of topic and sentiment in a document [18, 15, 12]: topics are modeled as simple mixtures under sentiment labels. It renders erroneous posterior inference results that assign opposite sentiment labels to the same topical aspects in a document. This inevitably leads to suboptimal performance in downstream sentiment analysis tasks.

In this work, we propose to explicitly model topic coherence and sentiment consistency in an opinionated text document so that we can accurately extract latent aspects and corresponding sentiment polarities. Specifically, we introduce hidden Markov model into topic modeling and name our solution as Hidden Topic Sentiment Model (HTSM). In HTSM, topics are modeled as a compound of latent aspects and sentiment polarities. Topic coherence is achieved by enforcing words in the same sentence to share the same topic assignment and modeling topic transition between successive sentences [8]. Sentiment consistency is imposed by constraining topic transitions via tracking sentiment changes – once sentiment assignment changes, a new topic has to be sampled for the current sentence. Both topic transition and sentiment transition are guided by a parameterized logistic function based on the linguistic signals directly observable in a document, e.g., cosine similarity and POS tag overlapping between adjacent sentences. A customized forward-backward algorithm is developed to perform efficient posterior inference for HTSM. The model configuration, including both word distribution under topics and topic/sentiment transitions, is learned in a fully unsupervised manner via expectation maximization. The formalization of HTSM is so flexible that partially annotated documents, e.g., user-provided pros and cons, can be easily incorporated for more accurate model estimation.

Extensive experimentations are performed on four categories of product reviews crawled from both Amazon and NewEgg to validate the effectiveness of the proposed model. A set of state-of-the-art topic models for sentiment analysis are employed as baselines to compare the quality of learned topics, accuracy of sentiment classification, and utility of aspect-based contrastive summarization from our HTSM model.

As a summary, our contributions in this paper are as follows,

- We develop a unified topic model to explicitly capture topic coherence and sentiment consistency in opinionated text documents. It provides more accurate extraction of latent topics and sentiment polarities.
- Our flexible modeling assumption enables both unsupervised and semi-supervised estimation of model parameters.
- We performed extensive experiment comparisons on different data sets under various application scenarios. Promising performance confirms the value of modeling dependence between sentiment and topic in sentiment analysis.

2. RELATED WORK

The wide coverage of topics and abundance of opinions in social media make it a gold mine for discovering public opinions on all sorts of topics [22]. Significant research effort has been paid on building statistic topic models to mine user-generated opinion data. According to the notion proposed in Mimno and McCallum's work [19], we can categorize most of existing topic models for sentiment analysis as upstream models and downstream models. Upstream models assume that in order to generate a word in a document, one needs to first decide the sentiment polarity of this word and then sample the topic assignment for this word accordingly. In contrast, downstream models assume the sentiment label is determined by the topic assignment in parallel to the text content.

Our proposed solution falls into the category of upstream models. One typical upstream model is the Topic-Sentiment Model (TSM) proposed in [18]. TSM is constructed based on the pLSA model [9]: in addition to assuming a corpus consists of a set of latent topics with neutral sentiment, TSM introduces two additional sentiment models, one for positive and one for negative sentiment. A new concept called "theme" is introduced in TSM for document modeling, and a theme is modeled as a compound of these three components: neutral topic words, positive words and negative words, in each document. However, this kind of division cannot capture the interrelation between topic and sentiment, given a document is still modeled as an unordered bag of words; and TSM also suffers from the same problems as in pLSA, e.g., overfitting and can hardly generalize to unseen documents.

Several follow-up work tries to address the limitations of TSM from different perspectives. Based on the LDA model [3], Lin and He proposed a joint sentiment/topic model (JST) for sentiment analysis [15]. In JST, the combination of topic and sentiment is modeled as a Cartesian product between a set of topic models and sentiment models. Accordingly, each document exhibits distinct topic mixtures under different sentiment categories in JST. To improve topic coherence, Jo and Oh extended JST by enforcing words in a single sentence to share the same topic and sentiment label in their Aspect and Sentiment Unification Model (ASUM) [12]. Zhao et al. introduced the Maximum Entropy LDA model (MaxEnt-LDA) to control the sampling of words from a background topic, aspect-specific topics and opinion-specific topics in [31]. Both JST and ASUM strongly depend on sentiment seed words to differentiate different sentiment categories. MaxEnt-LDA depends on a set of manually labeled training sentences with background, aspect and opinion words to estimate the maximum entropy model beforehand. Moreover, the simple sentiment-topic mixture assumption prevents all the aforementioned models to recognize sentiment consistency, i.e., sampling the same aspect assignment under different sentiment categories in a document.

Downstream models reverse the generation assumption between sentiment labels and topic assignments, and provide some flexibility in modeling sentiment, e.g., continuous opinion ratings can also be modeled [17, 28, 25]. However, downstream models usually assume the sentiment labels are observable, and it thus limits their applications in sentiment analysis.

Another line of related work is introducing Markov model into topic modeling. Aspect-HMM model [2] combines pLSA with a hidden Markov model [23] to perform document segmentation over text streams. However, Aspect-HMM separately estimates topics in training set and depends on heuristics to infer the transitional relations between topics. HMM-LDA [7] distinguishes short-range syntactic dependencies from long-range semantic dependencies a-

mong the words in each document. But in HMM-LDA, only the latent variables for the syntactic classes are treated as a locally depended sequence, while latent topics are treated the same as in other topic models. Hidden Topic Markov Model (HTMM) [8] is the most similar model to ours. HTMM captures topic coherence by assuming words in one sentence share the same topic assignment and modeling topic transitions between successive sentences. However, HTMM loosely models the transition between topics as a binary relation: the same as the previous sentence's assignment or draw a new one with a certain probability. It ignores sentiment consistence in a document: when sentiment switches, the topic assignments should also switch. Our HTSM constrains topic transition via tracking sentiment changes; and linguistic cues directly observable from adjacent sentences are leveraged to guide topic and sentiment transitions.

3. METHODOLOGY

In this section, we describe the proposed Hidden Topic Sentiment Model and discuss how it captures topic coherence and sentiment consistence simultaneously within an opinionated text document. Efficient posterior inference is performed via a customized forward-backward algorithm, and Expectation-Maximization algorithm is utilized to estimate the model parameters in both unsupervised and semi-supervised settings.

3.1 Definition of Terminologies

We first specify the notations and definitions of aspect, sentiment and topic used in this paper. Denote a set of review text documents about a particular type of entities, e.g., product reviews, as $D = \{d_1, d_2, \ldots, d_{|D|}\}$, where each document d_i consists of m sentences. We assume there is a shared set of aspects that attract reviewers' interest; and they can be defined as follows:

Definition (Aspect) An aspect of a particular entity is characterized by a set of words, which present a semantically coherent theme of discussion. An aspect can be indexed by a discrete random variable taking value from $A = \{a_1, a_2, \ldots, a_{|A|}\}$. For example, words such as "price", "value", and "worth" describe the *price* aspect of a product.

Beside describing the aspects, users also express their personal attitudes toward those aspects in their review documents, e.g., favor *price* aspect or criticize *customer service* aspect in product reviews. The expressed attitude is defined as sentiment.

Definition (Sentiment) Sentiment represents a user's emotional feelings about a particular entity. It can be denoted by a discrete random variable taking value from $S = \{s_1, s_2, \ldots, s_{|S|}\}$, e.g., positive or negative. In text documents, sentiment can be determined from content words. For example, "love" and "wonderful" indicate positive sentiment, and "terrible" and "regret" indicate negative sentiment.

In this paper, topic is defined as a compound of latent aspect and sentiment polarity. For example, in tablet reviews, potential topics could include positive aspect about battery life and negative aspect about customer service. Formally, topic is defined as follows:

Definition (Topic) Topic is a compound of latent aspect and sentiment polarity in a given document collection. It can be represented as a discrete distribution over words in a fixed vocabulary. Words with high probabilities under a topic depict the corresponding aspect and sentiment.

Based on the above definitions, we strive to develop a probabilistic generative model to automatically identify topics, i.e., aspects and sentiment, from a collection of opinionated text documents. The model takes an unstructured text document as input and returns decomposed latent aspects and sentiment polarities as output. In the following sections, we will discuss the detailed model assumptions and specifications.

3.2 Hidden Topic Sentiment Model

From linguistic analysis perspective, document exhibits internal structure, where structural segments encapsulate semantic units that are closely related [13]. As a result, in the proposed Hidden Topic Sentiment Model (HTSM), we treat sentence as the basic structure unit and assume all the words in a sentence share the same topic (as illustrated in our motivating example in Figure 1). Based on this, HTSM drops the simple mixture assumption employed in conventional topic models [3, 9], and explicitly models topic transition in successive sentences via a first-order hidden Markov model. Because in HTSM a topic is modeled as a compound of latent aspect and sentiment polarity, two factors control the transition of topics. First, once the sentiment labels switch between two consecutive sentences, a topic has to be generated for the subsequent sentence with a different aspect assignment. This enforces *sentiment consistency*. Second, when the sentiment labels stay intact, two adjacent sentences are assumed to be highly related: the subsequent sentence will inherent the topic assignment from the previous sentence, or select a distinct one from a document-specific topic mixture with certain probability. This imposes *topic coherence*.

Formally, we assume there are K topics embedded in a given collection of review documents. A topic indexed by z_k has two components: a_k indicates aspect label and s_k indicates sentiment label, i.e., $z_k = (a_k, s_k)$. Topic z_k is specified as a multinomial distribution over a fixed vocabulary V, i.e., $\{p(w|\beta_k)\}_{w \in V}$, where β_k is the corresponding model parameter. To avoid overfitting, we impose a shared Dirichlet prior over β_k, i.e., $\beta_k \sim Dir(\eta)$. In this paper, to simplify our discussion, we only consider binary sentiment polarities in HTSM, i.e., $s_k = \{0, 1\}$. But HTSM is flexible enough to model multi-class sentiment polarities, e.g., five-star rating scales [21].

In a given document d, the document-level topic proportion θ_d is assumed to be drawn from a shared Dirichlet distribution [3], i.e., $\theta_d \sim Dir(\alpha)$. Among m sentences in d, each sentence t_i has N_i words and is associated with a topic z_i, which is sequentially drawn from a document-specific Markov chain. Because the aspect label and sentiment polarity of sentences are unobservable, we introduce two latent variables τ and ψ on each sentence to control the sampling of topics with respect to the topic coherence and sentiment consistency requirements. Specifically, τ_i and ψ_i are binary random variables indicating whether there is a sentiment switch and an aspect change on sentence t_i accordingly. Their combination determines topic transition: 1) when $\tau_i = 0$ and $\psi_i = 0$, t_i will inherent previous sentence's topic assignment; 2) when $\tau_i = 0$ and $\psi_i = 1$, a new topic z_i will be drawn from θ_d, with the constraint that $s_i = s_{i-1}$ and $a_i \neq a_{i-1}$; 3) when $\tau_i = 1$ and $\psi_i = 1$, a new topic z_i will be sampled from θ_d with the constraint that $s_i \neq s_{i-1}$ and $a_i \neq a_{i-1}$. The combination of $\tau_i = 1$ and $\psi_i = 0$ is not allowed in HTSM, because the sentiment consistency constraint enforces aspect change when sentiment is switched.

To capitalize on the linguistic features directly observable in document content, e.g., overlapped sentence content indicates intact topic assignments, we use parameterized logistic functions to define the generation probability of τ and ψ in each sentence. Aspect transition feature function $f_a(d, i)$ takes document d and sentence t_i as input, and outputs an l-dimensional feature vector describing aspect change. Accordingly, $f_s(d, i)$ generates a p-dimensional

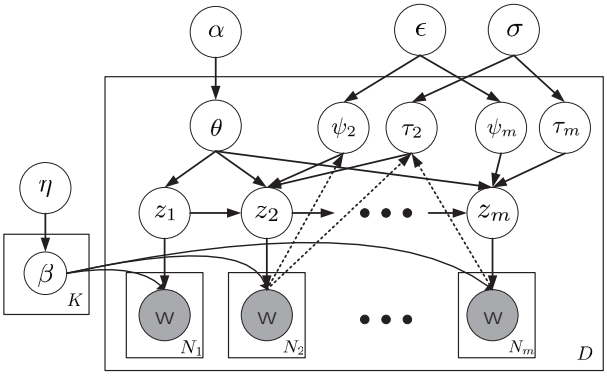

Figure 2: Graphical model representation of Hidden Topic Sentiment Model. Dark and light circles represent observable and latent random variables, and plates denote repetitions. Solid arrows encode dependency relation and dashed arrows denote the generation of transition features.

feature vector describing sentiment switch. Hence, we define,

$$p(\tau_i = 1|d, \sigma) = \frac{1}{1 + \exp\big(-\sigma^{\mathsf{T}} f_s(d, i)\big)} \quad (1)$$

$$p(\psi_i = 1|d, \epsilon) = \frac{1}{1 + \exp\big(-\epsilon^{\mathsf{T}} f_a(d, i)\big)} \quad (2)$$

where σ and ϵ are the corresponding feature weights for modeling sentiment switch and aspect change. The detailed specifications of $f_a(d, i)$ and $f_s(d, i)$ and the feature weight estimation procedures will be discussed in Section 3.4.

Putting above assumptions together, the generative process of a document postulated in HTSM can be described as follows,

1. For every topic z, draw $\beta_z \sim Dir(\eta)$.
2. For each review document $d \in D$,
   (a) Draw topic proportion $\theta_d \sim Dir(\alpha)$.
   (b) For each sentence $t_i \in d$, $i = 1, 2, \ldots, m_d$,
      i. Sample $\tau_i \sim p(\tau_i|d, \sigma)$; set $\tau_i = 1$ when $i = 1$.
      ii. Sample $\psi_i \sim p(\psi_i|d, \epsilon)$; set $\psi_i = 1$ when $\tau_i = 1$.
      iii. Sample z_i by,

$$z_i = \begin{cases} z_{i-1} & \text{if } \tau_i = 0, \psi_i = 0 \\ z \sim Mul(\theta_d), \textit{s.t. } a \neq a_{i-1}, s = s_{i-1} & \text{if } \tau_i = 0, \psi_i = 1 \\ z \sim Mul(\theta_d), \textit{s.t. } a \neq a_{i-1}, s \neq s_{i-1} & \text{if } \tau_i = 1, \psi_i = 1 \end{cases}$$

      iv. Sample each word w_n in t_i, $w_n \sim Mul(\beta_{z_i})$.

To make the above generation process consistent at every sentence in a document, we define $a_0 = \emptyset$ and $s_0 = \emptyset$, such that there is no constraint when sampling new topics for the first sentence in a document. Using the language of graphical models, this generation process can be visualized in Figure 2.

Conditioned on the model parameters $(\alpha, \beta, \epsilon, \sigma)$, the joint probability of sentences and latent topics in document d is thus given by,

$$p(\boldsymbol{z}, \theta, \boldsymbol{\psi}, \boldsymbol{\tau}, w_1, ..., w_{N_i}|\alpha, \beta, \epsilon, \sigma) \quad (3)$$
$$= p(\theta|\alpha) \prod_{i=1}^{m_d} p(\tau_i|d, \epsilon) p(\psi_i|d, \sigma) p(z_i|z_{i-1}, \tau_i, \psi_i, \theta) \prod_{n=1}^{N_i} p(w_n|\beta_{z_i})$$

The above joint distribution differentiates HTSM from conventional topic models for sentiment analysis, which are built on the simple topic mixture assumptions. Due to the sequential generation of topic assignments in sentences from a Markov chain, HTSM is no longer invariant to permutation of words nor sentences in a document. Documents in which successive sentences share coherent topics are more likely than any random shuffling of the same sentences. This leads to linearly coherent topic inference in a document: successive sentences tend to share similar topics, rather than fluctuated assignments. More importantly, sentiment consistency is especially emphasized in HTSM: in every sentence of a document, one needs to first determine if he/she wants to keep the sentiment polarity from previous sentence; if not, a new topic with different aspect label and sentiment polarity needs to be sampled. This avoids assigning contradictory sentiment polarities to the same aspect in a document. To the best of our knowledge, no existing topic models could achieve such regularity over topic assignments.

3.3 Posterior Inference

The latent variables of interest in HTSM are sentence-level topic assignments $\boldsymbol{z}$ and document-level topic proportion θ. The aspect switch indicators $\boldsymbol{\psi}$ and sentiment switch indicators $\boldsymbol{\tau}$ can be easily decoded from the topic assignment sequence $\boldsymbol{z}$. However, due to the coupling between continuous random variable θ and discrete random variables $\boldsymbol{z}$, exact inference in HTSM is computationally infeasible. In this paper, we develop a coordinate ascent based solution to perform approximate posterior inference.

In a given document d, θ can be first randomly initialized from its prior distribution $Dir(\alpha)$. With known θ, exact inference for $(\boldsymbol{z}, \boldsymbol{\psi}, \boldsymbol{\tau})$ can be efficiently performed via the forward-backward algorithm [23]. Because of the special design in our Markov chain, customization of the generic forward-backward algorithm can be made to greatly reduce its computational complexity in HTSM.

In particular, we treat the combination of (z_i, ψ_i, τ_i) at sentence t_i as latent states in our Markov chain for document d, and derive the corresponding transition function as,

$$p(z_i, \psi_i, \tau_i|z_{i-1}, \theta, d, \epsilon, \sigma) = p(z_i|z_{i-1}, \theta, \psi_i, \tau_i) \quad (4)$$
$$p(\psi_i|d, \epsilon) p(\tau_i|d, \sigma)$$

in which $p(\psi_i|d, \epsilon)$ and $p(\tau_i|d, \sigma)$ can be pre-computed beforehand since they are invariant during inference. And the first term of right-hand side in Eq (4) has a simple linear structure,

$$p(z_i|z_{i-1}, \theta, \psi_i, \tau_i) \propto \quad (5)$$
$$\begin{cases} 1 & \text{if } \tau_i = 0, \psi_i = 0, z_i = z_{i-1} \\ \theta_{z_i}, \textit{s.t. } a_i \neq a_{i-1}, s_i = s_{i-1} & \text{if } \tau_i = 0, \psi_i = 1 \\ \theta_{z_i}, \textit{s.t. } a_i \neq a_{i-1}, s_i \neq s_{i-1} & \text{if } \tau_i = 1, \psi_i = 1 \\ 0 & \text{otherwise} \end{cases}$$

This enables us to maintain a blockwise transition matrix in d and reduce the quadratic computational complexity in standard forward and backward computations to linear in HTSM.

After one round of forward-backward computation, posterior of θ_d can be computed by the expected frequency of words assigned to a topic that is drawn from the document-specific topic proportion, rather than inherited from a previous sentence. More specifically,

$$\theta_{d,z} \propto \sum_{i=1}^{m_d} \sum_{n=1}^{N_i} p(z_i = z, \psi_i = 1|d) + \alpha_z - 1 \quad (6)$$

The inference of θ and $(\boldsymbol{z}, \boldsymbol{\psi}, \boldsymbol{\tau})$ can be alternatively performed in a given document. And it can be proved that this coordinate ascent method will converge to a local maximum of data likelihood function in d, because the forward-backward algorithm gives us exact posterior of $(\boldsymbol{z}, \boldsymbol{\psi}, \boldsymbol{\tau})$ (refer to the EM algorithm proof [5]).

3.4 Parameter Estimation

Motivated by the insights gained from the annotated example shown in Figure 1, in HTSM we leverage content features directly observable in documents to define the probabilities of aspect

change and sentiment switch. In order to differentiate the aspect-driven transitions from sentiment-drive transitions, two sets of transition features are constructed.

The aspect transition features $f_a(d, i)$ include: 1) content-based cosine similarity between t_i and t_{i-1}; 2) sentence length ratio between t_i and t_{i-1}; 3) relative position of t_i in d, i.e., i/m; and 4) an indicator function about whether t_i is more similar to t_{i-1} or t_{i+1}. The sentiment transition features $f_s(d, i)$ include: 1) content-based cosine similarity between t_i and t_{i-1}; 2) sentiWordNet [1] score difference between t_i and t_{i-1}; 3) sentiment word count difference between t_i and t_{i-1}; 4) Jaccard coefficient between POS tags in t_i and t_{i-1}, and 5) adversative conjunction count in t_i. We also add bias terms in $f_a(d, i)$ and $f_s(d, i)$ to capture unconditioned aspect and sentiment transitions in documents. Detailed descriptions of these transition features can be found in Table 3.

The feature weights ϵ and σ in the transition functions defined in Eq (1) and Eq (2) can be efficiently estimated together with the other model parameters in HTSM by EM algorithm. In this work, we treat α and η as hyper-parameters of the model and manually tune their settings, given they have considerably less influence in model fitting [24] comparing to the other parameters, i.e., $(\beta, \epsilon, \sigma)$. We should note optimizing α and η with respect to data likelihood [3] is also feasible in HTSM.

The EM algorithm executes iteratively between E-step (for posterior inference) and M-step (for expectation maximization). In E-step at iteration T, the approximate inference procedures developed in Section 3.3 is executed in each document with the current model parameter $(\beta^T, \epsilon^T, \sigma^T)$. The following sufficient statistics are collected in documents after inference,

$$E[c(z, w, d)] = \sum_{i=1}^{m_d} \sum_{n=1}^{N_i} \delta(w_n = w) p(z_i = z|d) \quad (7)$$

$$E[\psi_i] = p(\psi_i = 1|d), \textit{s.t. } i > 1 \quad (8)$$

$$E[\tau_i] = p(\tau_i = 1|d), \textit{s.t. } i > 1 \quad (9)$$

In M-step, maximum likelihood estimator is used to compute $(\beta^{T+1}, \epsilon^{T+1}, \sigma^{T+1})$ as follows,

$$\beta_{z,w}^{T+1} \propto \sum_{d}^{|D|} E[c(z, w, d)] + \eta_w - 1 \quad (10)$$

$$\epsilon^{T+1} = \arg\max_{\epsilon} \sum_{d}^{|D|} \sum_{i=1}^{m_d} E[\psi_i] \log p(\psi_i = 1|d, \epsilon) \quad (11)$$

$$\sigma^{T+1} = \arg\max_{\sigma} \sum_{d}^{|D|} \sum_{i=1}^{m_d} E[\tau_i] \log p(\tau_i = 1|d, \sigma) \quad (12)$$

where the optimization of ϵ and σ can be effectively solved via a gradient-based optimizer. The E-step and M-step will be alternatively executed until the data likelihood function on the whole collection D converges.

In some review data sets, external signals about sentiment polarities are directly available. For example, some reviewers will explicitly organize their reviews in pros and cons sections [1]; and in NewEgg (`http://www.newegg.com/`), reviewers are required to do so. Such signals can be easily incorporated in HTSM to refine model estimation. In the documents with identified pros/cons sections, sentences in **pros** section will be considered as having sentiment label $s = 1$, and sentences in **cons** section will have $s = 0$. During posterior inference, the sentiment switch indicator τ can be directly computed from the sentiment labels in such documents, while all the rest inference steps stay the same. Hence, model parameter estimation in M-Step will be affected by such direct observations. As a result, HTSM effectively exploits such side information in document content and estimate the model parameters in a semi-supervised manner. In our quantitative evaluation, such semi-supervised model training greatly improves HTSM's sentiment classification performance.

[1] `http://www.amazon.com/gp/customer-reviews/R12HYQYZX5TNT9`

4. EXPERIMENT

In this section, we perform experiment evaluations of the proposed HTSM model from both quantitative and qualitative perspectives. We compare HTSM with several state-of-the-art topic models for sentiment analysis on four different collections of product reviews from both Amazon and NewEgg.

4.1 Data Sets & Preprocessing

We have collected four categories of product reviews, i.e., i) camera, ii) tablet, iii) tv and iv) phone, from Amazon (`http://www.amazon.com`) and NewEgg (`http://www.newegg.com`). The reviews from NewEgg are segmented into pros and cons sections by their original authors, since this is required by the website. The complete data set can be found at `http://www.cs.virginia.edu/~hw5x/dataset.html`.

Standard pre-processing is performed before the subsequent experiments. Firstly, punctuation, numbers and other non-alphabet characters are removed. Stopwords are also removed based on a standard stopword list [14]. Secondly, all the words are converted to the lower cases and stemming is performed on the remaining words in a document using the Porter's stemmer [30]. Finally, all the reviews which have less than five words are removed. Besides, since we are modeling topic transition between successive sentences, those reviews containing less than two sentences are also removed. Table 1 summarizes the resulting review data sets.

Table 1: Statistics of evaluation data sets.

Data set	Amazon	NewEgg	Vocabulary size	Positive ratio
camera	6919	3020	1406	0.606
tv	4729	1662	1410	0.551
tablet	6147	407	1515	0.494
phone	6899	268	1282	0.530

For comparison purposes, we include Latent Dirichlet Allocation (LDA) [3], Hidden Topic Markov models (HTMM) [8], Aspect and Sentiment Unification model (ASUM) [12], and Joint Sentiment/Topic model (JST) [15] as baselines. Among these baseline models, ASUM and JST are specialized for sentiment analysis, and HTMM and ASUM explicitly model sentences in a document. As unsupervised topic models, both ASUM and JST require sentiment seed words as input. Following the settings in their original paper, two sets of sentiment seed words are used in our experiments. The first one is from Turney's PARADIGM [26] contains seven positive words and seven negative words, and the second one is PARADIGM+ which contains all Turney's paradigm words plus other sentiment words. To conduct a fair comparison, we also include those sentiment seed words in our HTSM model, i.e., adding positive seed words to topics with sentiment label $s = 1$, and negative words to topics with sentiment label $s = 0$ as priors. We should note that unless otherwise specified, we have used 26 topics for camera and phone, 30 topics for tablet and 16 topics for tv for all the models. In addition, we fixed the hyper-parameters α and η in Dirichlet priors to 1.01 and 1.001 for all the topic models.

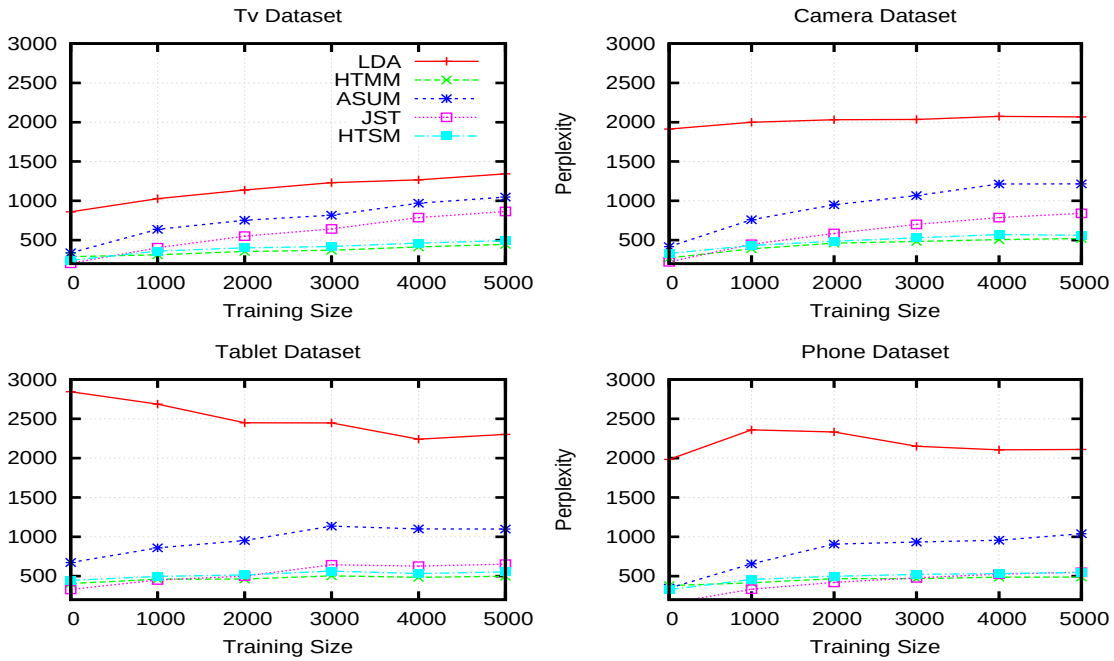

Figure 3: Perplexity with increasing training size on four different review document sets.

4.2 Topic modeling evaluation

We first compare the quality of learned topics from all the topic models. Perplexity and word intrusion experiments are performed to quantitatively evaluate this aspect, and we also demonstrate the learned topical transition diagram from HTSM.

4.2.1 Perplexity comparisons

Perplexity, used by convention in language modeling, is monotonically decreasing with respect to the likelihood of test data, and is algebraically equivalent to the inverse of the geometric mean of per-word likelihood. A lower perplexity indicates better generalization performance. More specifically, the perplexity of test document set D_{test} can be computed as:

$$perplexity(D_{test}) = \exp\left\{ -\frac{\sum_{d=1}^{M} \log p(w_d)}{\sum_{d=1}^{M} N_d} \right\} \qquad (13)$$

where M is the total number of documents in test corpus and N_d is the total number of word in a test document D_{test}.

We trained all the topic models (HTSM, HTMM, LDA, JST and ASUM) on the described corpora to compare their generalization performance in modeling text documents on a held-out test set via the perplexity measurement. Since our goal is to evaluate the density estimation quality, all documents in the corpora are treated as unlabelled (e.g., ignore the pros/cons segmentation in NewEgg reviews). The detailed experiment setup for perplexity comparison is as follows: we start with a training set containing only the reviews from NewEgg, refer this training set as the origin in plots of Figure 3, and gradually add more training reviews from Amazon (training size 1000, 2000 etc.). This experiment setting is to make the results aligned with the later sentiment classification experiments. Figure 3 demonstrates the average perplexity from five-fold cross validation (test sets are selected from both Amazon and NewEgg reviews accordingly). It is clear from Figure 3 that HTSM outperformed all the other topic models on all four datasets, except HTMM. There are two possible explanations. First, HTMM models topic transitions loosely as a Bernoulli distribution: the same as the previous sentence's assignment or drawing a new topic with certain probability. But HTSM models this topical transition with a more complicated logistic function. Overfitting might be caused by this parametric model. Second, HTMM does not consider sentiment in a document, i.e., less constraints in modeling a document. But in HTSM, once the sentiment label switches, a different topic has to be sampled for the subsequent sentence. As a result, HTMM has more freedom to allocate words under one topic, which results in a lower perplexity in modeling unseen documents. We should note that the perplexity metric only measures the quality of estimated word distribution in unseen documents. It cannot assess the sentiment prediction quality, which HTMM is unable to achieve. In later experiments we found that the increased complexity in HTSM benefits sentiment classification greatly. Finally, we can find that the simple sentiment-topic mixture assumptions in both JST and ASUM fail to capture the topic-word distribution in the test set and lead to much worse perplexity than HTSM.

It is also important to investigate how a topic model's generalization capability varies under different number of topics. In particular, we test different models' perplexity at the last testing point in Figure 3, i.e., five-fold cross validation on 5000 Amazon reviews with NewEgg reviews for training. Due to space limit, we only demonstrate the perplexity result from our HTSM model on all four categories of reviews in Figure 4. The baseline models exhibit similar patterns. From the results, it is clearly to observe that within a reasonable range of topic size, the perplexity of HTSM increases moderately. When we have more than 40 topics, the perplexity increases dramatically on all data sets, i.e., an indication of overfitting. The results justified our setting of the number of topics in HTSM and all baseline topic models, and we fix this setting in all our following experiments.

4.2.2 Word intrusion comparisons

Perplexity only measures the quality topic modeling from density estimation perspective; it is also necessary to evaluate whether

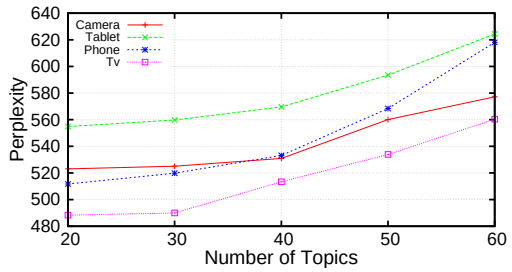

Figure 4: Perplexity of HTSM under different number of topics across all four categories of reviews.

the topics identified by those statistical models are human interpretable. More specifically, we prefer a model that generates more semantically coherent and meaningful topics.

In this experiment, we employ **word intrusion** discussed in [4] to evaluate four different topic models, namely LDA, HTMM, ASUM and HTSM (because ASUM and JST are quite similar in model assumptions, we do not include JST in this experiment). During the first phase of evaluation, our experiment setup is as follows: we first selected the top five words from each topic z_k under every model as topical words. Then we select two intruding words. The first intruding word is referred to as *intra-topic intrusion word*, which has a very low probability in topic z_k of corresponding models. The second intruding word is referred to as *inter-topic intrusion word*, which is selected from a different topic $z_l \neq z_k$ and has a high probability in topic z_l but a very low probability in topic z_k. To select a word which is considered as having a very low generation probability, we rank all the words under a topic in a descending order with respect to $p(w|z)$ and then randomly select a word with rank between 90 to 100 (given our vocabulary size on all collections is around 1400). Hence, in total we have seven words for each topic z_k from every topic model: among those, five are regular words, one is intra-topic intruding word and the last one is inter-topic intrusion word.

In the second phase of this evaluation, we randomly shuffled the topical words with the intruding words under each topic from every model and present the shuffled words to three annotators. The annotators do not have any knowledge about which topics or words have been generated by which model, and they are only informed of the category of the product. The task of the annotators is to identify at least one and at most two intruding words under each topic presented to them. In order to reduce annotation bias, we evenly separate the learned topics from each model into two parts, and present them to different annotators. We ensure that each topic is annotated by three different annotators. Since we have four different categories and four different topic models, for this task we take feedback from twenty four annotators. The agreement among annotators was calculated by pairwise Kappa statistics [27] and then these kappa values were averaged across all pairs of annotators. For example, on tablet data set, the average kappa value for original topical words is 0.885, which indicates annotators agree with each other most of time. However, for the intra-topic intrusion words and inter-topic intrusion words the average kappa values are 0.533 and 0.386 respectively, which imply that annotators might have different ways of interpreting the inferred topics.

To quantitatively measure the quality of inferred topics from all these four models, we define a metric named model word-intrusion recall (MR) as follows:

$$MR^m = \sum_{k=1}^{K} \frac{\sum_{s=1}^{S} \mathbf{1}(i^m_{z_k,s}, w^m_{z_k})}{K * S} \tag{14}$$

Table 2: Word intrusion measurement across different topic models of four categories of product reviews.

		Inter-topic MR		
Category	LDA	HTMM	ASUM	HTSM
camera	0.167	0.218	0.218	**0.282**
tablet	0.356	0.256	0.244	**0.389**
phone	0.192	0.179	0.231	**0.333**
tv	0.188	0.188	0.271	**0.313**
		Intra-topic MR		
Category	LDA	HTMM	ASUM	HTSM
camera	**0.474**	0.385	0.436	0.346
tablet	0.478	**0.533**	0.456	0.522
phone	**0.551**	0.500	0.487	0.346
tv	0.625	**0.646**	0.563	0.500

where $w^m_{z_k}$ is the vocabulary index of the intruding word among the words generated from the z^{th}_k topic inferred by topic model m, $i^m_{z_k,s}$ is the corresponding index of the intruding word selected by annotator s. S denotes the number of annotators, and K denotes the total number of topics.

From Table 2, it is evident that annotators can interpret the topics inferred by HTSM more effectively than those from the other models in terms of *inter-topic intrusion word*. For example, out of 90 actual inter-topic intrusion words in tablet category, 35 words have been picked out by annotators from HTSM's topics. This empirical evidence implies that our HTSM model is inferring more human interpretable topics than other topic models. However, in terms of *intra-topic intrusion*, the performance of HTSM is not as competitive as other models. The procedure of selecting low probability intra-topic intrusion word and the concentration of the learned word distribution under topics from HTSM might be contributing factors to the relative inferior performance of HTSM.

4.2.3 *Topic transitions*

Given HTSM explicitly models topic transitions in an opinionated review document, we visualize the learned transition using a transitional diagram to qualitatively demonstrate the topical coherence obtained by HTSM. Due to space limit, we only report the results extracted from tablet data set.

First, we train an HTSM with 30 topics on all the reviews from tablet category. To automatically differentiate domain-specific sentiment polarity, we train HTSM in a semi-supervised mode: the pros/cons sections in NewEgg reviews will be used to specify sentiment labels on sentences; while Amazon reviews will be used in fully unsupervised training. Then, for each sentence t_i in a review document d from training set, we infer its most probable topic z_k from HTSM via the Viterbi algorithm. As a result, for two consecutive sentences t_{i-1} and t_i, we have the corresponding pairwise topic transition $z_j \rightarrow z_k$. We accumulate the transition count based on all the consecutive sentences in the training corpus, and normalize the resulting transition matrix to construct the diagram.

Figure 5 illustrates the learned topic transition diagram in the tablet category. It is to be noted that in order to get a more perceivable view, we have ignored the transitions with probabilities less than 0.01 and also removed less popular topics in it. In this figure, each topic is denoted as a pair of *Aspect_Sentiment*. For example, *screen_P* represents positive sentiment about the *screen* aspect. In this transition diagram, there is also a special node named **start**, which is used to represent a dummy topic, which "generates" the initial topic for the first sentence in every document. Besides, we also highlighted the top six words under some selected topics (the selection of annotated topics is purely based on space constrain-

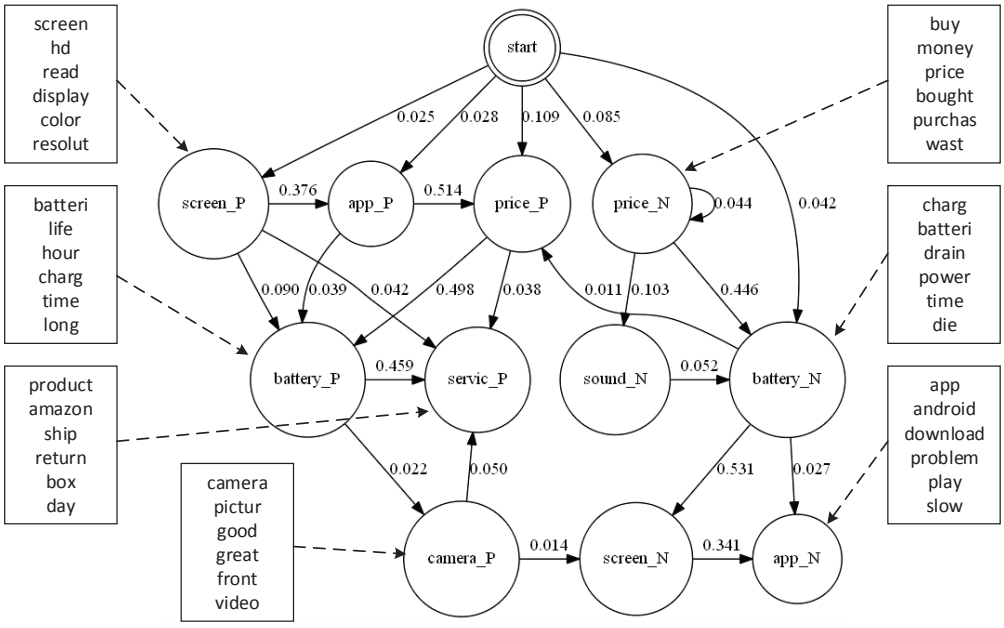

Figure 5: Estimated topic transition and top words under selected topic on tablet data set.

t). From Figure 5, we can clearly identify some interesting topical transitions in tablet reviews. For example, when reviewers hold positive feeling about their purchased tablets, they usually start with positive sentiment about "*price*," which is followed by positive sentiment about "*battery life*", "*service*" and so on. However, if a reviewer plans to criticize a tablet, he or she usually starts with negative sentiment about "*price*" and then transits to negative sentiment about "*battery life*", "*screen*", "*app*", and etc. This learned transition is of particular importance in opinion summarization: it helps organize the generated sentences in a coherent order.

4.3 Sentiment classification

In this section, we evaluate HTSM in terms of sentiment classification. We use the already segmented NewEgg reviews as ground-truth sentence-level sentiment annotations: we treat all sentences in the **pros** section as positive and all sentences in the **cons** section as negative. We should note such annotations are different from the overall ratings of reviews. The overall ratings are of low resolution in sentiment annotation: a review with high overall rating might still contain some negative sentences, and vice versa. In contrast, the self-annotated pros/cons sections are with finer-granularity in sentiment annotations. Therefore, in this experiment we did not use the overall ratings in model training and testing.

During the training phase of HTSM, we use a mixture of review data sets obtained from NewEgg and Amazon. Besides, since we have sentiment labels on sentences from the NewEgg data set, the sentiment transition indicator τ can be directly inferred. Hence we train our HTSM model in a semi-supervised manner. Specifically, during the training phase of HTSM, if the input document is from NewEgg, τ is fixed based on the sentiment labels on sentences; otherwise, HTSM has to infer τ according to Eq (1). To make a fare comparison across all the models, ASUM and JST were also modified to utilize the annotated pros/cons sections in NewEgg data set during the training phase. In addition, we also include EM-NaiveBayes [20], a semi-supervised algorithm, as a baseline in this experiment. It exploits the sentiment annotation in NewEgg data during the training phase.

We use only NewEgg data set to construct the test set, since we do not have such fine-grained annotations in Amazon data set (so we refer Amazon data as unlabelled data). Besides, we start our training set containing only the reviews from NewEgg (training size 0 in Figure 6) and then keep adding more and more unlabelled data from Amazon (training size 1000, 2000 etc.) into the training set, i.e., the exact setting that we used in perplexity evaluation in Section 4.2.1. We report the average F-1 score from five-fold cross-validation as the performance metric in this experiment.

Figure 6 illustrates the sentiment classification performance of HTSM over all the four categories against ASUM, JST and EM-NaiveBayes baselines. We can clearly notice that with the same amount of training data, HTSM outperformed all the other models, which treat sentences as independent in a document. Sentiment consistency enforced by HTSM helps to capture the dependence between consecutive sentences better and therefore predicts their sentiment polarities more accurately. The only exception is in the tv category, where the performance of HTSM degenerated beyond training size 3000 and became worse than EM-NaiveBayes. This degenerated result is caused by the divergent products reviewed in the Amazon and NewEgg data sets. We manually checked the products in tv category from these two data sets and found there are less common products than other categories. As a result, adding more Amazon reviews increases the discrepancy of the learned model on testing set, which is only from NewEgg reviews.

The improved classification performance of HTSM results from its unique capability in modeling sentiment consistency inside a review document, i.e., when sentiment switches, topic assignments have to change in successive sentences. The transitions are controlled by the parameterized logistic functions on the observable linguistic features described in Section 3.4. In Table 3, the learned feature weights for topic switch ϵ and sentiment switch σ on camera data set are demonstrated (we have very similar results on the other three categories as well, but due to space limit we cannot list them in the table). For example, the *bias* term controlling sentiment switch is more negative than that for topic transition. It implies that sentiment in two consecutive sentences are less likely to change than the topics. The learned weights for the content-based cosine

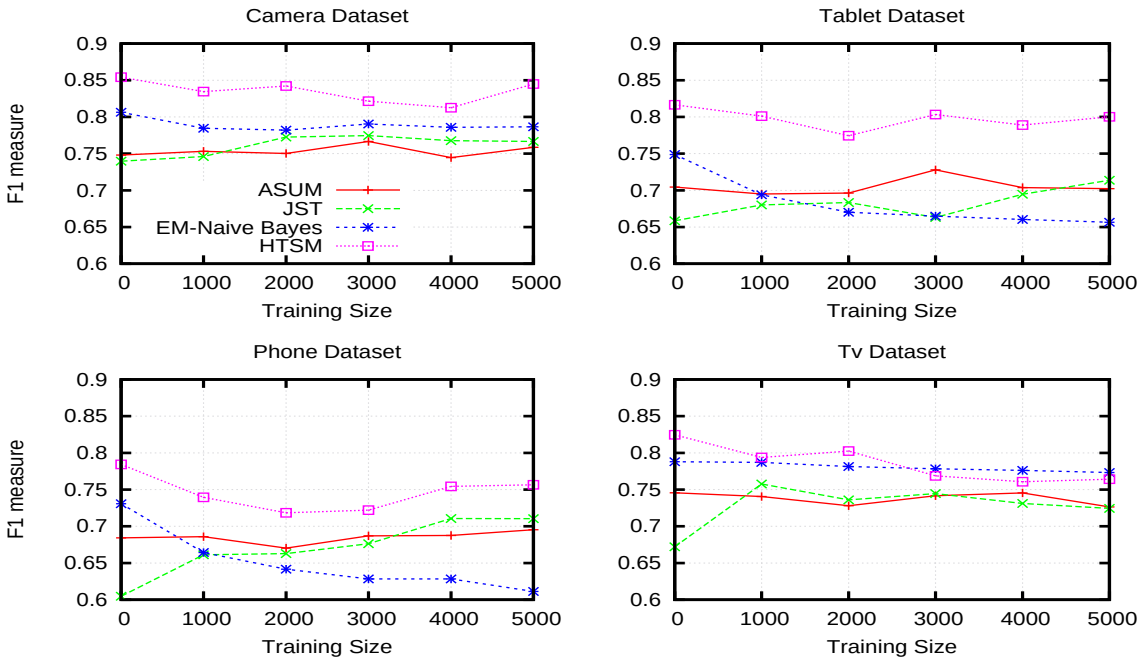

Figure 6: Sentiment classification performance with increasing training size on four different review document sets.

Table 3: Learned feature weights in HTSM for sentiment and topic transition on camera data set.

Sentiment transition feature	**Weight**
bias term $f_s(d, i)$	-2.271
content-based cosine similarity between t_i and t_{i-1}	-0.393
sentiWordNet [1] score difference between t_i and t_{i-1}	0.106
sentiment word count difference in t_i and t_{i-1}	0.251
an indicator function about whether t_i is more similar to t_{i-1} or t_{i+1}	0.521
jaccard coefficient between POS tags in t_i and t_{i-1}	0.049
negation word count in t_i	0.104
Topic transition feature	**Weight**
bias term $f_a(d, i)$	-0.016
content-based cosine similarity between t_i and t_{i-1}	-0.895
length ratio of two consecutive sentences t_i and t_{i-1}	0.034
relative position of t_i in d, i.e., i/m	0.225
an indicator function about whether t_i is more similar to t_{i-1} or t_{i+1}	0.233

similarity are negative for both transitions. It follows our expectation that the more similar two consecutive sentences are, the less likely we will observe sentiment or topic switch. These kind of observations well support our decision of using observable linguistic features to guide topic transition modeling and it ultimately helps HTSM to achieve improved topic coherence and sentiment consistency in modeling opinionated documents.

To provide a thorough evaluation of sentiment classification, we also tested all the topic models with varied number of topics. Following the same settings as in Figure 4, we reported the F1 measure of HTSM under all four categories of reviews. Due to space limit, we did not include the results from the baselines in Figure 7. Similar conclusion as that in perplexity evaluation can be reached: with a moderate number of topics in HTSM, its classification performance is satisfactory and stable; but with an increased number of topics, the classification results varied and even degenerated on some data sets (e.g., tablet data set).

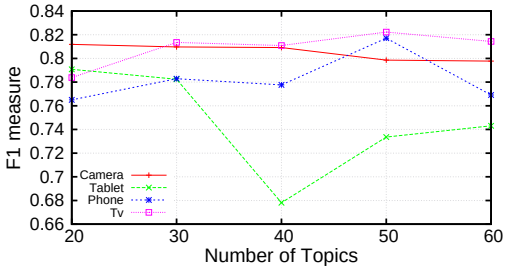

Figure 7: Sentiment classification performance of HTSM under different number of topics across all four categories of reviews.

4.4 Aspect-Based Contrastive Summarization

In order to evaluate the utility of the aspects and sentiments identified by our model, we study aspect-based review summarization, which aims at finding the most representative sentences for each topic (a combination of aspect and sentiment) from a collection of reviews. In Table 4, we demonstrate a sample aspect-based contrastive summarization result for two comparable tablet products. We selected *Samsung Galaxy Note 10.1* and *Amazon Kindle Fire HDX* based on their popularity in Amazon tablet data set. The practical value of this type of contrastive review summaries is to help customers easily digest vast amount of opinionated data and make informed decisions.

Table 4 shows the side-by-side comparison on six aspects ('+' indicates positive aspects and '-' indicates negative aspects) of these two tablets identified by HTSM. Imagine a user is making a choice between these two tablets. If the user cares battery aspect the most, he or she can easily find out from the summary that *Samsung Galaxy Note 10.1* is a better choice than *Amazon Kindle Fire HDX* by consulting this aspect-based contrastive review summarization. This saves the user considerable amount of time in reading the detailed reviews.

We perform user studies to understand whether these kind of summaries are meaningful for the actual users. In this experiment,

Table 4: Aspect-based contrastive summarization on tablet dataset.

Topic	Samsung Galaxy Note 10.1	Amazon Kindle Fire HDX
(+, battery)	Battery life is very good, it is easily an all day device with wifi on and high brightness while taking notes	Battery life is ok - probably need to recharge every other day with normal use
(-, battery)	My only issue is that it takes a long time to take a full charge and does not charge rapidly enough to use while charging, but the battery life is not bad	Everything works great, but the battery life is not nearly as long as advertised
(+, sound)	it has pretty good battery life, it also has an excellent quality sounding speakers, which i wasn't expecting on any tablet	Sound is really good (not home theater quality or anything) but better than any phone I've heard.
(-, sound)	The audio became occasionally inoperative and the headphone jack would crackle when using my ear buds	Users can get confused with volume buttons on the other side
(+, cpu)	quad core processor runs everything quickly and smoothly	The device features a fast 2.2GHz quad-core processor and 2GB of RAM for fast that run apps, games, and videos smoothly without an issues.
(-,cpu)	The OS was fast at first but as I added Apps it got slower and choppy	Compared to a galaxy note which is the same price, the Kindle HDX seems to have a slower processor

Table 5: Interleaved document summarization quality test.

Category	ASUM	HTMM	HTSM
camera	0.078	0.362	**0.560**
tablet	0.153	0.370	**0.477**
phone	0.118	0.439	**0.443**
tv	0.173	0.338	**0.489**

we choose three different topic models, including HTMM, ASUM and HTSM, given they all explicitly model sentences. We select the top two most probable sentences under each topic from every selected topic models, i.e., rank by $p(t|z)$. Since there are many different products under each category, we select three mostly reviewed products from each category for this user study. Once we have all sentences generated from those three topic models, we randomly interleave those sentences (to avoid position bias when annotating the results) and present them to the human annotators. The annotators do not have any knowledge of which sentence is picked by which model. Only the product name, the selected aspect and the corresponding sentiment are presented to the annotators (as we shown in Table 4). They were asked to pick one or two sentences which provide the most useful information about the presented aspect along with the required sentiment. In total, we recruited six annotators for each of the four categories. Inter-annotator agreement rate on document summarization quality judgment is calculated based on the pairwise Kappa statistic and then averaged across all pairs of annotators. For example, the average kappa value on tablet data set is 0.511 and on tv data set is 0.554. We define the following metric to evaluate the quality of generated summaries,

$$MS^m = \frac{\sum_{s=1}^{S}\sum_{k=1}^{K} N_s^{m,z_k}}{\sum_{m=1}^{M}\sum_{s=1}^{S}\sum_{k=1}^{K} N_s^{m,z_k}} \quad (15)$$

where N_s^{m,z_k} is the number of sentences picked by annotator s and selected by model m for topic z_k, S is the total number of annotators, K is the total number of selected topics and M is the total number of models.

Table 5 represents the summarization quality obtained by different models. We can find that annotators tend to select more sentences from HTSM than the other models as informative summaries. For example, in the tablet category, the annotators have selected in total 227 sentences, out of which 147 sentences are generated by HTSM. Comparing to HTMM, although both models impose transitional structure between consecutive sentences, HTMM cannot identify opinionated sentences in the generated summary. Comparing to ASUM, although it models sentiment polarities in sentences, the independence assumption limits its recognition of topical aspects in sentences. Therefore, based on this experimental evidence, we can conclude that quantitatively, HTSM is selecting sentences of better quality in both aspect recognition and sentiment polarities than the other topic models for aspect-based contrastive summarization, which is of particular value for customers to digest the opinionated information and make wise decisions.

5. CONCLUSION

In this paper, we present a unified generative model which jointly models sentiment and aspect in opinionated text documents. The proposed Hidden Topic Sentiment Model (HTSM) captures topic coherence by constraining the topic transition via tracking sentiment changes and utilizes the linguistic cues directly observable from adjacent sentences to guide topic and sentiment transitions. In contrast to the traditional sentiment-topic models which are built on simple topic mixture assumptions, HTSM captures the dependency between consecutive sentences by modeling document structure with a Markov assumption. Because of the sequential generation of topic assignment from a Markov chain, HTSM is no longer invariant to permutation of words or sentences in a document. Besides, sentiment consistency is strictly encoded in HTSM's transition modeling. Such properties enable HTSM to capture rich structure embedded in natural text documents. Extensive experiments have been performed to compare the performance of HTSM against several state-of-the-arts topic models on four categories of product reviews from Amazon and NewEgg. Improved topic modeling quality and sentiment classification performance are achieved.

This work opens new direction in topic modeling for sentiment analysis. The current HTSM only captures the first order Markov dependency among consecutive sentences, i.e. the current sentence is influenced only by the previous one. We can incorporate long-term dependency into HTSM, e.g., skip-chain, with controllable computational complexity. In addition, the semi-supervised training of HTSM depends on the availability of sentence-level annotations. It is important to incorporate document-level sentiment annotations in model training, e.g., utilize the companion overall numerical opinion ratings.

6. ACKNOWLEDGMENTS

We thank the anonymous reviewers for their insightful comments. This paper is based upon work supported in part by a Yahoo Academic Career Enhancement Award and the National Science Foundation under grants IIS-1553568.

7. REFERENCES

[1] S. Baccianella, A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *in Proc. of LREC*, 2010.

[2] D. M. Blei and P. J. Moreno. Topic segmentation with an aspect hidden markov model. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 343–348. ACM, 2001.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[4] 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*, pages 288–296, 2009.

[5] 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)*, pages 1–38, 1977.

[6] 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*, pages 63–72. ACM, 2012.

[7] T. Griffiths, M. Steyvers, D. Blei, and J. Tenenbaum. Integrating topics and syntax. *Advances in neural information processing systems*, 17:537–544, 2005.

[8] A. Gruber, Y. Weiss, and M. Rosen-Zvi. Hidden topic markov models. In *International Conference on Artificial Intelligence and Statistics*, pages 163–170, 2007.

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

[10] E. H. Hovy. Automated discourse generation using discourse structure relations. *Artificial intelligence*, 63(1):341–385, 1993.

[11] W. Jin, H. H. Ho, and R. K. Srihari. A novel lexicalized hmm-based learning framework for web opinion mining. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 465–472. Citeseer, 2009.

[12] 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*, pages 815–824. ACM, 2011.

[13] H. Kamp. A theory of truth and semantic representation. *Formal methods in the study of language*, 1:277–322, 1981.

[14] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li. Smart stopword list, 2004.

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

[16] J. McAuley and J. Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 165–172. ACM, 2013.

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

[18] 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*, pages 171–180. ACM, 2007.

[19] D. Mimno and A. McCallum. Topic models conditioned on arbitrary features with dirichlet-multinomial regression. *The 24th Conference on Uncertainty in Artificial Intelligence*, pages 411–418, 2008.

[20] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using em. *Machine learning*, 39(2-3):103–134, 2000.

[21] 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*, pages 115–124. Association for Computational Linguistics, 2005.

[22] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135, 2008.

[23] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[24] M. Steyvers and T. Griffiths. Probabilistic topic models. *Handbook of latent semantic analysis*, 427(7):424–440.

[25] I. Titov and R. T. McDonald. A joint model of text and aspect ratings for sentiment summarization. In *ACL*, volume 8, pages 308–316. Citeseer, 2008.

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

[27] A. J. Viera, J. M. Garrett, et al. Understanding interobserver agreement: the kappa statistic. *Fam Med*, 37(5):360–363, 2005.

[28] H. Wang, Y. Lu, and C. Zhai. Latent aspect rating analysis without aspect keyword supervision. In *Proceedings of the 17th ACM SIGKDD Conference*, pages 618–626. ACM, 2011.

[29] H. Wang, D. Zhang, and C. Zhai. Structural topic model for latent topical structure analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1526–1535. Association for Computational Linguistics, 2011.

[30] P. Willett. The porter stemming algorithm: then and now. *Program*, 40(3):219–223, 2006.

[31] 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*, pages 56–65. Association for Computational Linguistics, 2010.