

# Dynamic Topic Modeling for Monitoring Market Competition from Online Text and Image Data

Hao Zhang  
Carnegie Mellon University  
Pittsburgh, PA, 15213  
hao@cs.cmu.edu

Gunhee Kim  
Seoul National University  
Seoul, South Korea, 151-744  
gunhee@snu.ac.kr

Eric P. Xing  
Carnegie Mellon University  
Pittsburgh, PA, 15213  
epxing@cs.cmu.edu

## ABSTRACT

We propose a dynamic topic model for monitoring temporal evolution of market competition by jointly leveraging tweets and their associated images. For a market of interest (*e.g. luxury goods*), we aim at automatically detecting the latent topics (*e.g. bags, clothes, luxurious*) that are competitively shared by multiple brands (*e.g. Burberry, Prada, and Chanel*), and tracking temporal evolution of the brands' stakes over the shared topics. One of key applications of our work is social media monitoring that can provide companies with temporal summaries of highly overlapped or discriminative topics with their major competitors. We design our model to correctly address three major challenges: multi-view representation of text and images, modeling of competitiveness of multiple brands over shared topics, and tracking their temporal evolution. As far as we know, no previous model can satisfy all the three challenges. For evaluation, we analyze about 10 millions of tweets and 8 millions of associated images of the 23 brands in the two categories of *luxury* and *beer*. Through experiments, we show that the proposed approach is more successful than other candidate methods for the topic modeling of competition. We also quantitatively demonstrate the generalization power of the proposed method for three prediction tasks.

## Categories and Subject Descriptors

H.2.8 [Information Systems]: Database Applications—*Data mining*; G.3 [Probability and Statistics]: Probabilistic Algorithms; J.4 [Computer Applications]: Social and behavioral sciences—*Economics*

## Keywords

Dynamic topic models; Market competition; Text and images

## 1. INTRODUCTION

The increasing pervasiveness of the Internet has lead to a wealth of consumer-created data over a multitude of online

platforms such as blogs, discussion forums, and social networking sites. Such contents are valuable for companies to listen in consumers' candidate opinions, and thus there have been many recent studies on *online market intelligence* [10, 17, 18], whose goal is collecting and analyzing online information that is contributed by the general public toward companies' products and services, and providing with pictures of ongoing brand performance in a set of given market conditions. The online market intelligence has been one of emerging fields in data mining research as market competition becomes fierce, and consumers' online reviews and evaluations are considered more trustworthy and spontaneous than other information described by vendors.

In this paper, we address the problem of modeling temporal evolution of market competition by jointly leveraging text data and their associated image data on the Web. More specifically, we study tweets and their linked images. Fig.1 illustrates the problem statement of this paper. For a specified competitive market (*e.g. luxury goods*), multiple brands (*e.g. Burberry, Chanel, and Rolex*) compete one another to raise their stakes over shared values or topics, which include products-related topics such as *bags, clothes, and watch*, or consumers' sentiments-related topics such as *luxurious, expensive*. The objective of this research is to build an automatic system that crawls tweets, extract text and images from tweets, identify shared topics that multiple brands compete to possess one another, and track the evolution of brands' proportional dominance over the topics.

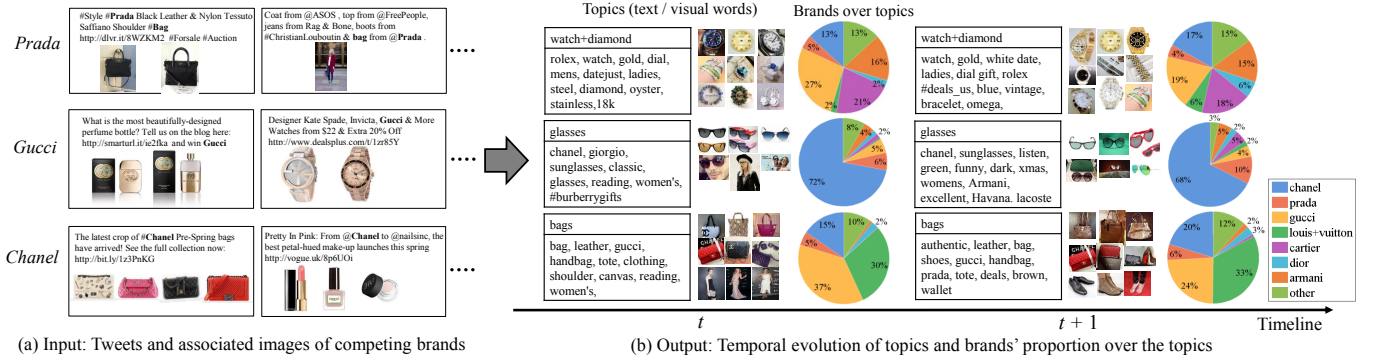
Our approach focuses on the joint analysis of text and image data tagged with the names of competing brands, which have not been explored yet in the previous studies of online market intelligence. The joint interpretation of text and images is significant for several reasons. First, a large portion of tweets simply show images or links without any meaningful text in them. Hence, images play an important role for representing topics in this type of tweets. In our dataset, 70% of tweets are attached with urls, and 28% of tweets in the *luxury* category are with images. Second, many users prefer to use images to deliver their idea more clearly and broadly, and thus the topic detection with images reflects users' intents better. The popularity of images can be seen in a simple statistics of our twitter dataset; our *luxury* corpus contains more images than tweets (*e.g. 5.5 millions tweets with 6.6 millions of images*). Third, the joint use of images with text also helps marketers interpret the discovered topics. Due to the short length of tweets (*i.e. 140 characters*), marketers may need to see the associated images to understand key ideas of tweets easier and quicker. Finally, since

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [Permissions@acm.org](mailto:Permissions@acm.org).

KDD'15, August 10-13, 2015, Sydney, NSW, Australia.

© 2015 ACM. ISBN 978-1-4503-3664-2/15/08 ...\$15.00.

DOI: <http://dx.doi.org/10.1145/2783258.2783293>.



**Figure 1: Problem statement.** (a) Input is a large collection of tweets and their associated images that are retrieved by the names of competing brands in a market of interest. (b) As output we aim at identifying the topics that are shared by multiple brands, and track the evolution of topics and proportion of brands over the topics.

the Internet is where users cannot physically interact one another about actual products or services, images may be essential for users to make conversation about customers' descriptions, experiences, and opinions toward the brands.

From technical viewpoints, we propose a novel dynamic topic model to correctly address the following three major challenges: (1) multi-view representation of text and images, (2) modeling of latent topics that are competitively shared by multiple brands, and (3) tracking temporal evolution of the topics. Some of existing work attain a subset of these challenges (*e.g.* texts and images [4, 7] and dynamic modeling [1, 5]), but none of them satisfies all the challenges.

We evaluate our algorithm using newly collected dataset from *Twitter* from October 2014 to February 2015. Our automatic crawler downloads all tweets tagged by brand names of interest, along with attached or linked images if available. Consequently, our dataset contains about 10 millions of original tweets and 8 millions of associated images of the 23 brands in the two categories of *luxury* and *beer*. The experiments demonstrate the superior performance of the proposed approach over other candidate methods, for dynamic topic modeling and three prediction tasks including prediction of the most associated brands, most-likely created time, and competition trends for unseen tweets. Note that while we mainly deal with brands of the two categories, our approach is completely unsupervised and thus applicable, without any modification, to any categories once input sets of text and image streams are collected.

The foremost application of our work is social media monitoring, which assists marketers to summarize their fans' online tweets with sparse and salient topics of competition in an illustrative way. Especially, our algorithm can discover and visualize the temporal progression of what topics are highly *overlapped* or *discriminative* over other competitors. From our interaction with marketers, we observe that they are very curious to see and track what topics emerge and what pictures their fans (re-)tweet the most, but there is no such system yet. As another application, our method can be partly used for sentiment analysis [17] because the detected topics can be positive or negative. That is, multiple brands competes one another not only on positive topics (*e.g.* multiple cosmetics brands compete on the *health+beauty* topic) but also negative topics (*e.g.* multiple beer brands compete on the *drunk+driving* topic). We do not perform in-depth analysis on sentiment analysis because it is out of the scope, but at least marketers can observe their brands' distribution

on both positive and negative topics, which is also useful for market analysis. Although we mainly focus on the applications of brand competitions in a market, our problem formulation and approach are much broader and are applicable to other domains of competition, including tourism (*e.g.* multiple cities compete to attract more international tourists), and politics (*e.g.* multiple candidates contest to take leads on major issues to win an election), to name a few.

The main contributions of this paper are as follows. (1) To the best of our knowledge, our work is the first attempt so far to propose a principled topic model to discover the topics that are competitively shared between multiple brands, and track the temporal evolution of dominance of brands over topics by leveraging both text and image data. (2) We develop a new dynamic topic model for market competition that achieves three major challenges of our problem; multi-view representation of text and images, modeling of competitiveness of multiple entities over shared topics, and tracking their temporal evolution. As far as we know, no previous model can satisfy all the challenges. (3) With experiments on more than 10 millions of tweets with 8 millions of images for 23 competing brands, we show that the proposed algorithm is more successful for the topic modeling over other candidate methods. We also quantitatively demonstrate the generalization ability of the proposed method for three prediction tasks.

## 2. RELATED WORK

**Online Market Intelligence.** One of most closely related line of work to ours is *online market intelligence* [17], whose objective is, broadly speaking, to mine valuable information for companies from a wealth of consumer-generated online data. Due to vast varieties of markets, brands, and information to mine, there have been many different directions to address the problem as follows. As one of early successful commercial solutions, the BRANDPLUSE platform [10] monitors consumers' *buzz* phrases about brands, companies, or any emerging issues from public online data. In [15], market-structure perceptual maps are automatically created to show which brands are jointly discussed in consumers' forums especially for the two categories of market, which are sedan cars and diabetes drugs. The work of [24] focuses on extracting comparative relations from Amazon customer reviews, and visualize the comparative relation map (*e.g.* *Nokia N95 has a better camera than iPhone*). The authors of [2] also leverage Amazon data to discover the relations between

product sales and review scores of each product feature (*e.g.* battery life, image quality, or memory for digital cameras). In [22], a recommendation system on the blogosphere is developed to learn historical weblog posts of users, and predict which users the companies need to follow when they release new products. Our work has two distinctive features over existing research of this direction. First, we address an unexplored problem of detecting the latent topics that are competitively shared by multiple brands, and automatically tracking their temporal evolution. Second, we jointly leverage two complementary modalities, text and images, which have been rare in market intelligence research.

**Topic Models for Econometrics.** Lately, there have been significant efforts to develop generative topic models for modeling and prediction of economic behaviors of users on the Web. In [8], a simple LDA model is applied to stock market data to detect the groups of companies that tend to move together. The work of [11] proposes a new dynamic topic model to predict the temporal changes of consumers' interests and purchasing probabilities over catalog items. In [13], a geo-topic model is developed to learn the latent topics of users' interests from location log data, and recommend new location that are potentially interesting to users. Finally, [14, 19] are examples of topic models that are applied to the tasks of opinion mining and sentiment analysis, in which they produce fine-grained sentiment analysis from user reviews or weblog posts. Compared to previous research of this direction, our problem of modeling market competition of multiple brands is novel, and our model is also unique as an econometric topic model that jointly leverages online texts and images.

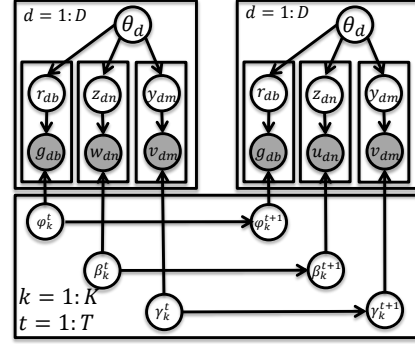
**Dynamic and Multi-view Topic Models.** There has been a large body of work to develop dynamic topic models to analyze data streams [8, 11, 13, 14, 19], and multi-view topic models to discover the interactions between text and images in multimedia contests [4, 7, 9, 21]. Compared to existing dynamic and multi-view topic models, our approach is unique in the ability of directly modeling the competition of multiple entities (*e.g.* brands) over shared topic spaces. Since previous models cannot handle with the interactions between multiple entities, they are only applicable to the dataset of each brand separately. However, in this case, the detected topics can be different in each brand; thus it is difficult to elicit shared topic spaces to model the competition.

### 3. A DYNAMIC MODEL FOR MARKET COMPETITION

We first discuss how to represent online documents and associated images, and then develop a generative model for market competition.

#### 3.1 Representation of Text and Images

Suppose that we are interested in a set of competing brands  $\mathcal{B} = \{1, \dots, \mathcal{B}^L\}$  in the same market (*e.g.* Chanel, Gucci, and Prada as *luxury* brands). We use  $\mathcal{B}^l$  to denote a set of documents (*i.e.* tweets) that are downloaded by querying brand name  $l$  in the time range of  $[1, T]$ . We assume that each document  $d \in \mathcal{B}^l$  consists of text and optionally URLs that are linked to images. That is, a tweet can be text only or associated with one or multiple images. Some tweets may be associated with multiple brand labels, if they are retrieved multiple times by different brand names. We use a



$\phi_k^t$	Brand-topic occupation matrix at time $t$ ( $\in \mathbb{R}^{K \times L}$ )
$\beta^t / \gamma^t$	Topic distributions over text/visual words at time $t$ ( $\in \mathbb{R}^{K \times G} / \mathbb{R}^{K \times H}$ ).
$\theta_d$	Document code of document $d$ ( $\in \mathbb{R}^K$ ).
$z_{dn} / y_{dm}$	Word code of text/visual word $n/m$ ( $\in \mathbb{R}^K$ ).
$w_{dn} / v_{dm}$	Occurrences of text/visual word $n/m$ in document $d$ .
$r_{db}$	Brand code of brand $b$ in document $d$ ( $\in \mathbb{R}^K$ ).
$g_{db}$	Indicator for each brand label $b$ for document $d$ .

**Figure 2: Plate diagram for the proposed topic model with a table of key random variables.**

vector  $\mathbf{g}_d \in \mathbb{R}^L$  to denote which brands are associated with document  $d$ .

For the text descriptor, we use the TF-IDF weighted *bag of words* model [4], where we build a dictionary of text vocabularies after removing words occurred fewer than 50 times. For image descriptor, we leverage ImageNet pre-trained deep learning features with vector quantization. Specifically, we use Oxford VGG MatConvnet and utilize their pre-trained model CNN-128 [20]<sup>1</sup>, which a compact 128-dimensional descriptor for each image. Then, we construct  $H$  *visual clusters* by applying  $K$ -means clustering to randomly sampled (at max) two millions of image descriptors. We assign the  $r$ -nearest visual clusters to each image with the weights of an exponential function  $\exp(-a^2/2\sigma^2) + \epsilon$ , where  $a$  is the distance between the descriptor and the visual cluster,  $\sigma$  is a spatial scale, and  $\epsilon$  is a small positive value to prevent zero denominator when normalization. Finally, each image is described by an  $H$  dimensional  $\ell_1$  normalized vector with only  $r$  nonzero weights. In our experiments, we set  $H = 1,024$ ,  $\sigma = 10$ , and  $r = \|\mathbf{u}\|_0$  which is the  $\ell_0$ -norm of its corresponding text descriptor, so that text and image descriptors have the same number of nonzeros.

As a result, we can represent every document and image as a vector. If we let  $U = \{1, \dots, G\}$  and  $V = \{1, \dots, H\}$  to denote sets of vocabularies for text and visual words respectively, each document  $d$  is represented by a pair of vector  $(\mathbf{u}_d, \mathbf{v}_d)$ , where  $\mathbf{u}_d = [u_{d1}, \dots, u_{d|N|}]^T$  where  $N$  is the index set of words in document  $d$ , and each  $u_{dn}$  ( $n \in N$ ) represents the number of appearances of word  $n$ . Likewise,  $\mathbf{v}_d = [v_{d1}, \dots, v_{d|M|}]^T$  where  $M$  is the index set of visual words. If a document has multiple associated images,  $\mathbf{v}_d$  is represented by a vector sum of image descriptors. For a document with no associated image,  $\mathbf{v}_d$  becomes a null vector and  $M$  is an empty set.

#### 3.2 A Probabilistic Generative Process

<sup>1</sup>[http://www.robots.ox.ac.uk/~vgg/software/deep\\_eval/](http://www.robots.ox.ac.uk/~vgg/software/deep_eval/).

Our model is designed based on our previous *Sparse Topical Coding* (STC) framework [26], which is a topic model that can directly control the posterior sparsity. In our problem setting, each document and word is encouraged to be associated with only a small number of strong topics. Since we aim at analyzing the possibly complex interaction between multiple brands, in practice a few salient topical representation can make interpretation easier rather than letting every topic make a non-zero contribution. In addition, the sparsity leads a more robust text/image representation since most of tweet documents are short and sparse in word spaces due to length limitation of 140 characters. Another practical advantage of the STC is that it supports simultaneous modeling of discrete and continuous variables such as image descriptors and brand associations.

However, our model significantly extends the STC in several aspects. First, we update the STC to be a *dynamic* model so that it handles the streams of tweets. Second, we extend to jointly leverage two complementary information modalities, text and associated images. Finally, we address an unexplored problem of detecting and tracking the topics that are competitively shared by multiple brands. All of them can be regarded as novel and nontrivial improvement of our method.

Fig.2 shows the graphical model for the proposed generative process. Let  $\beta \in \mathbb{R}^{K \times G}$  and  $\gamma \in \mathbb{R}^{K \times H}$  be the matrices of  $K$  topic bases for each text and visual word respectively. That is,  $\beta_k$  indicates the  $k$ -th text topic distribution over the vocabularies  $U$ . We also use  $\phi \in \mathbb{R}^{K \times L}$  to denote the brand-topic occupation matrix, which expresses the proportions of each brand over topics. We denote  $\theta_d \in \mathbb{R}^K$  as the *document code*, which is a latent topic distribution of document  $d$ .  $z_{dn} \in \mathbb{R}^K$  and  $y_{dm} \in \mathbb{R}^K$  are the *text/visual word code* respectively, which are latent topic representation of individual text word  $n$  and visual word  $m$  in document  $d$ .

Below we discuss in detail the generative process of our model, which is summarized in Table 1.

**Multi-view STC model.** For text content, we use the similar generative process with that of the original STC [26]. For each document  $d$ :

1. Sample a document code  $\theta_d \sim \text{prior } p(\theta)$ .
2. For each observed word  $n \in N$ ,
  - (a) Sample a word code  $z_{dn} \sim p(z|\theta_d)$ .
  - (b) Sample an observed word count  $u_{dn} \sim p(u|z_{dn}, \beta)$ .

In order to model documents with both text and images, we develop a multi-view extension. Specifically, for each document  $d$ , we let its text part  $u_d$  and its corresponding image part  $v_d$  share the same document code  $\theta_d$ , as shown in Fig.2. In addition, we assume the same generative process for visual words with the text counterpart. Consequently, we supplement the following step.

3. For each observed visual word  $m \in M$ ,
  - (a) Sample a visual word code  $y_{dm} \sim p(y|\theta_d)$ .
  - (b) Sample a visual word count  $v_{dm} \sim p(v|y_{dm}, \gamma)$ .

We now define the distributions used in the above process. Since each tweet is represented by a very sparse vector in a word space, the document code of a tweet is preferred to be sparse in a topic space in order to foreground the most salient topics and suppress noises. To achieve sparsity on  $\theta$ , we define the document code prior  $p(\theta)$  as a Laplacian prior  $p(\theta) \propto \exp(-\lambda \|\theta\|_1)$ , which becomes a  $\ell$ -1 regularizer

in the negative log posterior. Similarly, to boost the topical sparsity of each word, we define the conditional distributions of word codes as the following composite distribution:

$$\begin{aligned} p(z_{dn}|\theta_d) &\propto \exp(-\delta_u \|z_{dn} - \theta_d\|_2^2 - \rho_u \|z_{dn}\|_1) \\ p(y_{dm}|\theta_d) &\propto \exp(-\delta_v \|y_{dm} - \theta_d\|_2^2 - \rho_v \|y_{dm}\|_1), \end{aligned} \quad (1)$$

which establishes a connection between the document code and word codes while encouraging sparsity on the word codes.

For the last step of generating word counts, the STC recommends to use an exponential family distribution with the linear combination  $z_{dn}^\top \beta_{\cdot n}$  as a mean parameter to make optimization easier and the model applicable to rich forms of data. That is,  $\mathbb{E}_p[u] = z_{dn}^\top \beta_{\cdot n} + \epsilon$  where  $\beta_{\cdot n}$  denotes the  $n$ -th column of  $\beta$  and  $\epsilon$  is a small positive number for avoiding degenerated distributions. We choose to use a Gaussian distribution with the mean of  $z_{dn}^\top \beta_{\cdot n}$ , and apply the same idea to the visual word counts. Therefore,

$$\begin{aligned} p(u_{dn}|z_{dn}, \beta) &= \mathcal{N}(u_{dn}; z_{dn}^\top \beta_{\cdot n}, \sigma_u^2 I) \\ p(v_{dm}|y_{dm}, \gamma) &= \mathcal{N}(v_{dm}; y_{dm}^\top \gamma_{\cdot m}, \sigma_v^2 I). \end{aligned} \quad (2)$$

**Dynamic extension.** In order to model the temporal evolution of topics, we let  $\beta$  and  $\gamma$  to change over time, based on the discrete dynamic topic model (dDTM) [5]. That is, we divide a corpus of documents into sequential groups per time slice  $t$  (e.g. one week in our experiments), and assume that the documents in each group  $D^t$  are exchangeable. Then we evolve  $\beta^t$  and  $\gamma^t$  from the ones in previous time slice  $t-1$  by following the state space model with a Gaussian noise. Therefore, for each topic  $k$ , we use

$$\begin{aligned} p(\beta_k^t | \beta_k^{t-1}) &= \mathcal{N}(\beta_k^{t-1}, \sigma_\beta^2 I) \\ p(\gamma_k^t | \gamma_k^{t-1}) &= \mathcal{N}(\gamma_k^{t-1}, \sigma_\gamma^2 I). \end{aligned} \quad (3)$$

**Competition extension.** We now extend the multi-view dSTC to capture the competition between multiple brands over topics. We first define a brand-topic occupation matrix  $\phi \in \mathbb{R}^{K \times L}$  to represent the proportions of brands on latent topics. For each document  $d$ , we denote  $B \subseteq \mathcal{B}$  as the index set of brands, and  $g_d \in \mathbb{R}^B$  as an  $\ell$ -1 normalized vector representing associated brand labels. For example, if tweet document  $d$  is retrieved by keywords  $\{\text{prada}, \text{chanel}\}$ , then  $B = \{\text{prada}, \text{chanel}\}$  and  $g_d = [g_{d1} \ g_{d2}]$ , which are normalized values describing how strong the tweet is associated with the observed brands. One can use the same values (e.g.  $g_{d1} = g_{d2} = 0.5$ ) or proportional values according to relevance scores by the twitter search engine. For each  $b \in B$  and  $g_{db}$ , we use a latent brand code  $r_{db} \in \mathbb{R}^K$  as a representation of brand  $b$  in the topic space. We let  $r_{db}$  to be conditioned on the document code  $\theta_d$ , which governs the topic distributions of not only text/visual words but also brand labels.

There are two possible options of dynamics on the brand-topic occupation matrix  $\phi$ . First, similarly to  $\beta$  and  $\gamma$ , we evolve  $\phi$  to capture potential dynamics between brands and latent topics over the time. In this case, we use the state space model with a Gaussian noise, and thus  $\phi$  has the same distribution of Eq.(3). Second, if we assume that the brand occupation over topics is stationary, we can sample  $\phi$  from a uniform distribution. We take the first approach.

Finally, we can apply the same distributions to the generative process for the brands with the counterparts of text and visual words. That is, we use the composite distribution of Eq.(1) for  $p(r_{db}|\theta_d)$ , and the Gaussian distribution

of Eq.(2) for  $p(g_{db}|\mathbf{r}_{db}, \phi)$ . In summary,

$$\begin{aligned} p(\mathbf{r}_{db}|\boldsymbol{\theta}_d) &\propto \exp(-\delta_b \|\mathbf{r}_{db} - \boldsymbol{\theta}_d\|_2^2 - \rho_b \|\mathbf{r}_{db}\|_1) \\ p(g_{db}|\mathbf{r}_{db}, \phi) &= \mathcal{N}(g_{db}; \mathbf{r}_{db}^\top \boldsymbol{\phi}_b, \sigma_b^2 I) \\ p(\phi_k^t | \phi_k^{t-1}) &= \mathcal{N}(\phi_k^{t-1}, \sigma_\phi^2 I). \end{aligned} \quad (4)$$

## 4. LEARNING AND INFERENCE

In this section, we describe the optimization for learning and inference of the proposed model.

### 4.1 MAP Formulation

The generative process of Fig.2 provides a joint probability for a document  $d$  in each time slice  $t$ :

$$p(\boldsymbol{\theta}, \mathbf{z}, \mathbf{u}, \mathbf{y}, \mathbf{v}, \mathbf{r}, \mathbf{g} | \boldsymbol{\beta}, \boldsymbol{\gamma}, \phi) = p(\boldsymbol{\theta}) \prod_{n \in N} p(\mathbf{z}_n | \boldsymbol{\theta}) p(u_n | \mathbf{z}_n, \boldsymbol{\beta}) \prod_{m \in M} p(\mathbf{y}_m | \boldsymbol{\gamma}) p(v_m | \mathbf{y}_m, \boldsymbol{\gamma}) \prod_{b \in B} p(\mathbf{r}_b | \phi) p(g_b | \mathbf{r}_b, \phi) \quad (5)$$

If we add the superscript  $t$  to explicitly represent the time slice for each variable, we can denote the parameter set as follows:  $\boldsymbol{\Theta}^t = \{\boldsymbol{\theta}_d^t, \mathbf{z}_d^t, \mathbf{y}_d^t, \mathbf{r}_d^t\}_{d=1}^{D^t}$ , where  $\mathbf{z}_d^t = \{\mathbf{z}_{dn}^t\}_{n \in N_d^t}$ ,  $\mathbf{y}_d^t = \{\mathbf{y}_{dm}^t\}_{m \in M_d^t}$  and  $\mathbf{r}_d^t = \{\mathbf{r}_{db}^t\}_{b \in B_d^t}$ , where  $N_d^t$  denotes the word index set of document  $d$  in time slice  $t$ , and likewise for  $M_d^t$  and  $B_d^t$ . Although we skip the derivation due to space limitation, it is not difficult to show the negative log posterior for time slice  $t$  satisfies

$$\begin{aligned} -\log p(\boldsymbol{\Theta}^t, \boldsymbol{\beta}^t, \boldsymbol{\gamma}^t, \phi^t | \{\mathbf{u}_d^t, \mathbf{v}_d^t, \mathbf{g}_d^t\}_{d=1}^{D^t}) \\ \propto -\log\{p(\boldsymbol{\Theta}^t, \{\mathbf{u}_d^t, \mathbf{v}_d^t, \mathbf{g}_d^t\}_{d=1}^{D^t} | \boldsymbol{\beta}^t, \boldsymbol{\gamma}^t, \phi^t)\}. \end{aligned} \quad (6)$$

In the above,  $\lambda, \{\nu_i, \delta_i, \pi_i\}_{i=1}^3$  are hypeparameters, which are chosen by cross validation in our experiments.

### 4.2 Parameter Estimation

We estimate the model parameters by minimizing the negative log posterior derived in previous section. Since Eq.(6) is the one for the documents in a single time slice  $t$ , we accumulate the negative log posteriors of all time ranges, and seek for an optimal solution for the whole corpus of all time slices. Therefore, the final objective is derived as

$$\begin{aligned} \min_{\{\boldsymbol{\Theta}^t, \boldsymbol{\beta}^t, \boldsymbol{\gamma}^t, \phi^t\}_{t=1}^T} \sum_{t=1}^T \sum_{d=1}^{D^t} \lambda \|\boldsymbol{\theta}_d^t\|_1 \\ + \sum_{t=2}^T (\pi_1 \|\boldsymbol{\beta}^t - \boldsymbol{\beta}^{t-1}\|_2^2 + \pi_2 \|\boldsymbol{\gamma}^t - \boldsymbol{\gamma}^{t-1}\|_2^2 + \pi_3 \|\phi^t - \phi^{t-1}\|_2^2) \\ + \sum_{t=1}^T \sum_{d=1}^{D^t} \sum_{n \in N_d^t} (\nu_1 \|\mathbf{z}_{dn}^t - \boldsymbol{\theta}_d^t\|_2^2 + \rho_1 \|\mathbf{z}_{dn}^t\|_1 + \mathcal{L}(\mathbf{z}_{dn}^t, \boldsymbol{\beta}^t)) \\ + \sum_{t=1}^T \sum_{d=1}^{D^t} \sum_{m \in M_d^t} (\nu_2 \|\mathbf{y}_{dm}^t - \boldsymbol{\theta}_d^t\|_2^2 + \rho_2 \|\mathbf{y}_{dm}^t\|_1 + \mathcal{L}(\mathbf{y}_{dm}^t, \boldsymbol{\gamma}^t)) \\ + \sum_{t=1}^T \sum_{d=1}^{D^t} \sum_{b \in B_d^t} (\nu_3 \|\mathbf{r}_{db}^t - \boldsymbol{\theta}_d^t\|_2^2 + \rho_3 \|\mathbf{r}_{db}^t\|_1 + \mathcal{L}(\mathbf{r}_{db}^t, \phi^t)) \end{aligned} \quad (7)$$

s.t.  $\boldsymbol{\theta}_d^t \geq 0, \forall d, t, \mathbf{z}_{dn}^t, \mathbf{y}_{dm}^t, \mathbf{r}_{db}^t \geq 0, \forall d, n, m, b, t,$

$$\boldsymbol{\beta}_k^t \in \mathcal{P}_U, \boldsymbol{\gamma}_k^t \in \mathcal{P}_V, \phi_k^t \in \mathcal{P}_B, \forall k, t,$$

where  $\mathcal{P}_U, \mathcal{P}_V, \mathcal{P}_B$  are the  $G-1, H-1$  and  $L-1$  simplex, respectively (i.e. For  $\forall k, t$ , each of  $\boldsymbol{\beta}_k^t, \boldsymbol{\gamma}_k^t$  and  $\phi_k^t$  should

For each time slice  $t$ :

1. Draw a text topic matrix  $\boldsymbol{\beta}^t | \boldsymbol{\beta}^{t-1} \sim \mathcal{N}(\boldsymbol{\beta}^{t-1}, \sigma_\beta^2 I)$ .
2. Draw an image topic matrix  $\boldsymbol{\gamma}^t | \boldsymbol{\gamma}^{t-1} \sim \mathcal{N}(\boldsymbol{\gamma}^{t-1}, \sigma_\gamma^2 I)$ .
3. Draw a brand topic matrix with two options: (i) dynamic  $\phi^t | \phi^{t-1} \sim \mathcal{N}(\phi^{t-1}, \sigma_\phi^2 I)$ , or (ii) independent  $\phi^t \sim \text{Unif}(0, 1)$ .
4. For each document  $d = (\mathbf{u}, \mathbf{v})$  in  $D^t$ ,
  - (a) Sample a document code  $\boldsymbol{\theta}_d \sim \text{prior } p(\boldsymbol{\theta})$ .
  - (b) For each observed text word  $n \in N$ ,
    - i. Sample a word code  $\mathbf{z}_{dn} \sim p(\mathbf{z}_{dn} | \boldsymbol{\theta}_d)$ .
    - ii. Sample a word count  $u_{dn} \sim p(u | \mathbf{z}_{dn}, \boldsymbol{\beta})$ .
  - (c) If  $M$  is not an empty set:
    - i. For each observed visual word  $m \in M$ ,
      - A. Sample a visual word code  $\mathbf{y}_{dm} \sim p(\mathbf{y}_{dm} | \boldsymbol{\theta}_d)$ .
      - B. Sample a visual word count  $v_{dm} \sim p(v | \mathbf{y}_{dm}, \boldsymbol{\gamma})$ .
  - (d) For each observed brand  $b \in B$ ,
    - i. Sample a latent brand code  $\mathbf{r}_{db} \sim p(\mathbf{r}_{db} | \boldsymbol{\theta}_d)$
    - ii. Sample a brand association  $g_{db} \sim p(g | \mathbf{r}_{db}, \phi)$

**Table 1: The generative process of the proposed model (See text for details).**

sum to one). We denote  $\mathcal{L}$  as the negative log-loss of reconstruction for word counts and brand associations in Eq.(2):

$$\mathcal{L}(\mathbf{z}_{dn}^t, \boldsymbol{\beta}^t) = -\log p(u_{dn}^t | \mathbf{z}_{dn}^t, \boldsymbol{\beta}^t) = \delta_1 \|\mathbf{u}_{dn}^t - \mathbf{z}_{dn}^{\top} \boldsymbol{\beta}_{.n}\| \quad (8)$$

Thanks to the use of an exponential family distribution for generating word counts (e.g. Gaussian distributions of Eq.(2)), the loss function  $\mathcal{L}$  is convex, and thus the optimization of Eq.(7) is multi-convex (i.e. the optimization is convex over one parameter set when the others are fixed). Consequently, we can directly employ coordinate descent to solve the optimization problem.

Taking into consideration that tweet documents grow along with time, we propose two approaches for solving the above problem, namely *online learning* and *smoothing*. The two approaches are similar except that the online learning seeks for a local minimum in the current time slice, based on the data of one or several previous time slices, which can be more scalable for online monitoring of real-world big data. On the other hand, the smoothing approach globally optimizes the objective over the data in all time slices, which is less scalable but yields more accurate fitness for data, and thus can be more suitable for batch analysis.

#### 4.2.1 Smoothing Approach

In the smoothing approach, we directly optimize the objective of Eq.(7). Note that every two adjacent time slices are only coupled by three parameters:  $\boldsymbol{\beta}, \boldsymbol{\gamma}$  and  $\phi$ . Hence, if we fix these three parameters, the objective for each time slice is independent one another. Based on this idea, we alternate between the optimization for  $\boldsymbol{\beta}, \boldsymbol{\gamma}, \phi$  and the one for the other variables using the coordinate descent algorithm:

1. Fix all  $\{\boldsymbol{\beta}^t, \boldsymbol{\gamma}^t, \phi^t\}_{t=1}^T$ . We now decouple the optimization of every time slice  $t$ . Since documents can be assumed to be independent one another, we further decou-

ple per document  $d$ . Therefore we solve

$$\begin{aligned}
& \min_{\theta_d^t, \mathbf{z}_d, \mathbf{y}_d, \mathbf{r}_d} \lambda \|\theta_d^t\|_1 \quad (9) \\
& + \sum_{n \in N_d^t} (\nu_1 \|\mathbf{z}_{dn}^t - \theta_d^t\|_2^2 + \rho_1 \|\mathbf{z}_{dn}^t\|_1 + \mathcal{L}(\mathbf{z}_{dn}^t, \beta^t)) \\
& + \sum_{m \in M_d^t} (\nu_2 \|\mathbf{y}_{dm}^t - \theta_d^t\|_2^2 + \rho_2 \|\mathbf{y}_{dm}^t\|_1 + \mathcal{L}(\mathbf{y}_{dm}^t, \gamma^t)) \\
& + \sum_{b \in B_d^t} (\nu_3 \|\mathbf{r}_{db}^t - \theta_d^t\|_2^2 + \rho_3 \|\mathbf{r}_{db}^t\|_1 + \mathcal{L}(\mathbf{r}_{db}^t, \phi^t)) \\
& s.t. : \theta_d^t \geq 0; \mathbf{z}_{dn}^t, \mathbf{y}_{dm}^t, \mathbf{r}_{db}^t \geq 0, \forall n.
\end{aligned}$$

Note that for every document  $d \in D^t$ , if  $\theta_d$  is fixed,  $\mathbf{z}_d$ ,  $\mathbf{y}_d$  and  $\mathbf{r}_d$  are independent one another. Thus, we can use the coordinate descent to alternately optimize  $\theta_d$  and  $\mathbf{z}_d, \mathbf{y}_d, \mathbf{r}_d$ .

- While fixing  $\theta_d$ , we solve each of  $\mathbf{z}_{dn}, \mathbf{y}_{dm}, \mathbf{r}_{db}$  independently, all of which have close-form solutions. Specifically, the solution for the  $k^{th}$  element of  $\mathbf{z}_{dn}^t$  is  $z_{dnk}^t = \max(0, \frac{u_{dn}^t \beta_{kn}^t + \sigma_1 \theta_{dk}^t - \beta_{kn}^t \sum_{j \neq k} z_{dnj}^t \beta_{jn}^t - \frac{\rho_1}{2}}{\beta_{kn}^t + \sigma_1})$ , where  $\sigma_1 = 1$ .
- While fixing  $\mathbf{z}_d, \mathbf{y}_d$ , and  $\mathbf{r}_d$ , we solve the following problem to update  $\theta_d$ :

$$\begin{aligned}
& \min_{\theta_d^t} \lambda \|\theta_d^t\|_1 + \sum_{n \in N_d^t} \nu_1 \|\mathbf{z}_{dn}^t - \theta_d^t\|_2^2 \quad (10) \\
& + \sum_{m \in M_d^t} \nu_2 \|\mathbf{y}_{dm}^t - \theta_d^t\|_2^2 + \sum_{b \in B_d^t} \nu_3 \|\mathbf{r}_{db}^t - \theta_d^t\|_2^2 \\
& s.t. \theta_d^t \geq 0
\end{aligned}$$

The optimal  $\theta$  of this problem is the truncated average of  $\mathbf{z}_{dn}^t, \mathbf{y}_{dm}^t, \mathbf{r}_{db}^t$  [26]. We drop the term including  $\mathbf{y}_{dm}^t$  for the documents with no associated image.

- While fixing all parameters of  $\{\Theta^t\}_{t=1}^T$ , we optimize

$$\begin{aligned}
& \min_{\{\beta^t, \gamma^t, \phi^t\}_{t=1}^T} \sum_{t=2}^T \sum_{d=1}^D \sum_{n \in N_d^t} \mathcal{L}(\mathbf{z}_{dn}^t, \beta^t) + \sum_{t=2}^T \pi_1 \|\beta^t - \beta^{t-1}\|_2^2 \\
& + \sum_{t=2}^T \sum_{d \in D^t} \sum_{m \in M_d^t} \mathcal{L}(\mathbf{y}_{dm}^t, \gamma^t) + \sum_{t=2}^T \pi_2 \|\gamma^t - \gamma^{t-1}\|_2^2 \\
& + \sum_{t=2}^T \sum_{d \in D^t} \sum_{b \in B_d^t} \mathcal{L}(\mathbf{r}_{db}^t, \phi^t) + \sum_{t=2}^T \pi_3 \|\phi^t - \phi^{t-1}\|_2^2 \\
& s.t. \beta_k^t, \gamma_k^t, \phi_k^t \in \mathcal{P}, \forall k, t \quad (11)
\end{aligned}$$

We can obtain the optimal of Eq.(11) by separately solving  $\{\beta^t\}_{t=1}^T, \{\gamma^t\}_{t=1}^T, \{\phi^t\}_{t=1}^T$ , because they are independent one another. When we solve each of them, for example of  $\{\beta^t\}_{t=1}^T$ , we utilize the coordinated descent and the projected gradient descent, in which we solve every  $\beta^t$  one by one for each  $t$ . That is, at every iteration we fix all  $\{\beta^t\}_{t=1}^T \setminus \beta^t$ , and use projected descent to solve  $\beta^t$ . We iterate until convergence for every  $t$ .

#### 4.2.2 Online Learning Approach

Instead of directly optimizing the objective of Eq.(7), the online learning approach assumes that at every time  $t$ , we

only observe a new set of data at  $t$ , and have learned model parameters from the data up to  $t-1$ . This can be more practical in a real-world scenario; we may not always globally optimize using all the past data when we observed new data. Instead we would better seek for a local minimal that may be good enough to reflect the current state of market competition. Formally, we assume that at each  $t$  we only consider its  $p$  previous time slices to form an evolving chain. To make our discussion easier, we set  $p=1$ ; however, it is not difficult to derive the optimization solver for  $p>1$ .

Given the optimization algorithm for smoothing approach in previous section, online learning optimization is readily straightforward. At time slice  $t$  we assume that we are given the MAP solutions up to  $t-1$ , which are denoted by  $\hat{\beta}^{t-1}$ ,  $\hat{\gamma}^{t-1}$ , and  $\hat{\phi}^{t-1}$ . We sample  $\beta^t$  from the distribution  $p(\beta^t | \hat{\beta}^{t-1})$  as defined in Eq.(3). We do the same for  $\gamma^t$  and  $\phi^t$  as well. Once we have  $\beta^t, \gamma^t$ , and  $\phi^t$ , as discussed in previous section, the objective of Eq.(7) for each time slice is independent one another. Thus, we can directly apply the algorithm presented in previous section to solve the decoupled objective in every time slice one by one along with time. At the start of the optimization, we initialize all  $\{\beta^t\}_{t=1}^T, \{\gamma^t\}_{t=1}^T, \{\phi^t\}_{t=1}^T$  using a uniform prior.

## 5. EXPERIMENTS

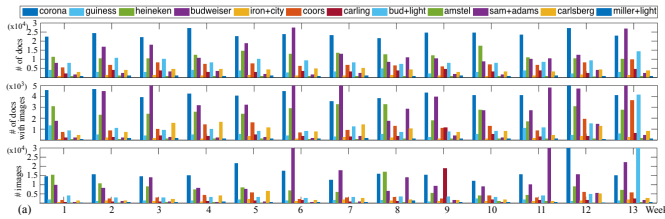
We evaluate our model from the following four aspects. First, we qualitatively and quantitatively evaluate the quality of learned text and visual topics (Section 5.2.1). Second, we show how our model can simultaneously monitor topic evolution and market competition along with time, compared to some baseline methods (Section 5.2.2). Third, we design three prediction methods based on our model, to show the generalization power of our model for unseen documents (Section 5.3). Finally, we conduct internal comparisons and provide some analysis on our model (Section 5.4).

### 5.1 Twitter Dataset for Multiple Brands

Fig.3 summarizes some statistics of our Twitter dataset for two groups of competing brands: *Luxury* and *Beer*. We query brand names using Twitter's RESTs API without any filtering, in order to obtain users' free and uncensored views on the brands. The data range from 10/20/2014 to 02/01/2015, during which our crawler is scheduled to run once every week, 3 days per week to finish the weekly crawling job. After obtaining raw tweets, we use a publicly available tokenizer for Twitter [16] to extract text and valid URLs from each tweet, and eliminate illegal, non-English characters, and stop words, while preserving emoticons, blocks of punctuation and twitter catchwords<sup>2</sup>. In addition, our crawler traverses every legal URL, and downloads images located in the body of HTML pages. We exclude the images that have too small file sizes or unreasonable aspect ratios.

We extract text and image descriptors as described in section 3.1. Note that our text and image descriptors for the same document have the same number of nonzero elements (i.e.  $|N| = |M|$ ). We then mean-align the two descriptors by setting  $mean(\mathbf{u}) = mean(\mathbf{v})$ . We standardize the text and image descriptor to avoid bias on any of them. For tweets with multiple images, we obtain the vector sum of all image descriptors, and standardize it.

<sup>2</sup>We follow [26] to use a standard list of 524 stop words.



**Figure 3: Statistics of our newly collected twitter dataset on the timeline. We report the numbers of (tweets, tweets with images, images) from top to bottom. (a) The *Beer* corpus = (1,091,369, 231,318, 829,207) (b) The *Luxury* corpus = (5,511,887, 935,903, 6,606,125).**

Consequently, the *Beer* corpus involves 12 brands, yielding 1,101,192 raw tweets and 829,207 images. We build a dictionary of 12,488 text vocabulary words after removing words occurred fewer than 50 times. Finally, we obtain 1,091,369 valid tweet documents, out of which 231,318 tweets contain images as well. The *Luxury* corpus is much larger than the beer corpus, including 5,572,017 raw tweets and 6,606,125 images. Following the same preprocessing step, we obtain a dictionary of 36,023 words, and 5,511,887 tweet documents and 1,523,177 ones associated with images.

## 5.2 Model Evaluation

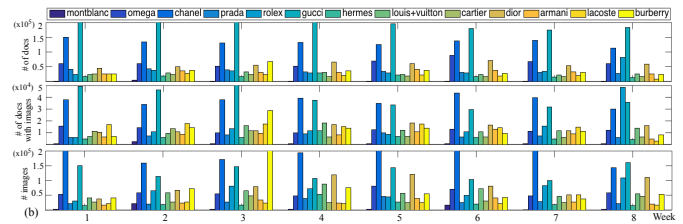
In this section, we evaluate the performance of topic detection and tracking of our model, and demonstrate its application to the market competition monitoring.

### 5.2.1 Topic Quality and Evolution

We assess the quality of the learned topics by our model, compared to other commonly used topic models. Our goal here is to quantitatively show that (1) our model captures the common semantics shared in the tweet corpus textually and visually. (2) Our approach successfully tracks the topic evolution along with time.

While it is still an open problem how to quantitatively evaluate topic models, perplexity and held-out likelihood have been popular measures to assess how well a topic model can be generalized to unseen documents. However, we do not use perplexity and held-out likelihood, because they are not a proper metric in our evaluation for the two following reasons. First, the work of [6] performs a large scale experiment on the Amazon Mechanical Turk, and suggests that the perplexity and human judgment are often not correlated. Second, more importantly, our preliminary experiments reveal that they are not fair metrics for the comparison between the algorithms that use different distributions in the model. For example, our model shows a perplexity 10 times lower than other methods, because we model text/visual word counts using Gaussian, which always leads a higher per-word likelihood than Multinomial distribution in LDA or Poisson regressor in STC.

Therefore, we quantitatively evaluate the coherence and validity of our learned topics by extending the Coherence Measure (CM) defined in [23], which is inspired by human evaluation methods of [6]. Specifically, for every text topic, we select the top 10 words with the highest probabilities. Then, we ask 10 human annotators to judge whether the 10 words can be understood as a single specific topic. If not, the topic is labeled as *ineffective*. The annotators are further asked to scan every word and classify it as *relevant* or *irrelevant* to the topic. If more than a half of words are classified as relevant, then the topic is regarded as *coherent*. Similarly, for each visual topic, we use the same evaluation



	VM ( <i>Beer</i> / <i>Luxury</i> )	CM ( <i>Beer</i> / <i>Luxury</i> )
dLDA	0.53 / 0.68	0.55 / 0.52
STC+dyn	0.44 / 0.66	0.57 / 0.57
cdSTC+multi	0.51 / 0.70	<b>0.63 / 0.59</b>
cdSTC+text	<b>0.605 / 0.71</b>	0.61 / <b>0.59</b>

**Table 2: Average VM/CM comparison on text topics.**

	VM ( <i>Beer</i> / <i>Luxury</i> )	CM ( <i>Beer</i> / <i>Luxury</i> )
KMeans	0.39 / 0.56	0.59 / 0.64
LDA+multi	<b>0.57</b> / 0.63	0.51 / 0.69
cdSTC+multi	<b>0.57</b> / <b>0.65</b>	<b>0.66</b> / <b>0.71</b>

**Table 3: Average VM/CM comparison on visual topics.**

strategy: we first provide the labelers with the top 10 visual words of a visual topic, each of which is represented by top 10 nearest images. The labelers are asked to scan all 100 images globally to judge whether they illustrate a single specific topic. If yes, the visual topic is labeled as *effective*. Then the labelers classify the images in the each visual word as related or unrelated with the topic, and more than a half of images are classified as related, then the visual word is regarded as coherent with this topic. Based on the user study results, we define the validity measure (VM) and coherence measure (CM), as two metrics of the topic quality:

$$VM = \frac{\# \text{ of valid topics}}{\# \text{ of topics}}, CM = \frac{\# \text{ of relevant words}}{\# \text{ of words in valid topics}}.$$

For experiments, we train the text-only and multi-view version of our model, **cdSTC+text** and **cdSTC+multi**, using the data of all time slices. We set the topic number to 50. For the tests of text topics, we compare with two baselines: (1) **dLDA** [5]: dynamic LDA, and (2) **STC+dyn** [26]: the STC trained using the data up to  $t - 1$  time slice. For tests of visual topics, we compare our results with two baselines: (1) **KMeans**: a simple baseline of k-means clustering. Specifically, we cluster the descriptor vectors of documents with images to 50 clusters, and regard each center as a topic, extract the nearest 10 images as an illustration of every center. (2) **LDA+multi**: A multi-view LDA implemented based on [12]. Following [5], we use the data at  $t - 1$  time slice for training.

Table 2 and Table 3 show the average *VM* and *CM* results rated by the 10 human annotators. For text topics, our **cdSTC+text** achieves the best results on the *VM* measure, compared to the **dLDA** and **STC+dyn** models. For visual topics, our **cdSTC+multi** attains the highest score, which concludes that joint use of text and images help detect more human-interpretable topics.

### 5.2.2 Monitoring Brand Competitions

In this section, we demonstrate the application of our model to the market competition monitoring. Given social



media data of multiple brands, our model can solve the following three tasks, from easy to difficult: (1) At one time slice, we monitor their occupations on latent topics. (2) Along the timeline, we monitor the trend of each brand's occupation over the topics. (3) Along the timeline, we monitor the global competition trends between multiple brands.

Fig.4 illustrates the evolving chain of topic *beauty* on the luxury corpus in eight time slices from 2014-10-20 to 2014-12-15. We also show the brand competition pie graphs and the trend curve of every brand occupation on the timeline. Our model successfully captures the topic dynamics; the *beauty* topic gradually evolves with time, from *makeup* and *lip* to *blackfriday*, *order* and *deals*, and finally steps into *winter*, involving more health-related words like *skin-care* and *hydra-potection*. The visual words also carry variations along with time, which are consistent with the text topics. The following eight pie graphs shows the competitions of the top seven brands on each time slice. We observe that (1) the *dior* dominates the *beauty* topic all the time and overwhelm the *gucci*, which is the largest brand that occupies almost a half of our whole corpus, and (2) other brands (*e.g.* *burberry*, *chanel*, *gucci*) show dynamic up-and-downs over the topics along with the time, which can be useful pieces of information for marketers.

### 5.3 Evaluation on Prediction

We further verify the generalization ability of the proposed model through three prediction tasks. The first two tasks are classification problems, which have been tested for evaluation in many topic model papers (*e.g.* [4, 7, 25, 26]). The third task helps marketers compare between interpolated trends and actual topic distribution side by side.

#### 5.3.1 Prediction of Associated Brands

The goal of the first prediction task is to estimate the most associated brand for a novel tweet. Although this task can be seen as a multi-class classification problem where a plenty of other methods can be applied, we perform this task to prove the generalization power of our model on unseen data. For this prediction, we make two modifications to our model. First, we drop the terms related to brand competitions, which are not required for classification. Second, we develop a supervised extension to be applicable to classification problems. We use the document code as the input of a multi-class max-margin classifier, and jointly train the latent representations and multi-class classifiers. The supervised dSTC (**sdSTC**) solves the following problem<sup>3</sup>:

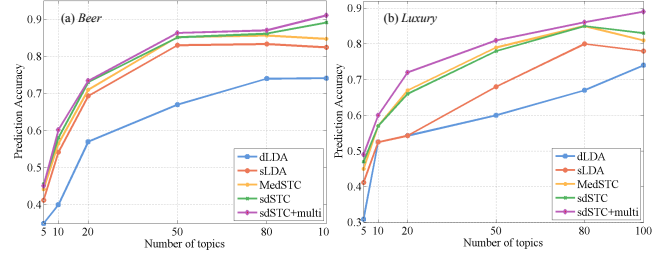
$$\begin{aligned} \min_{\{\Theta^t, \mathcal{M}^t, \eta^t\}_{t=1}^T} & \sum_{t=1}^T f(\Theta^t, \mathcal{M}^t, D^t) + CR(\Theta^t, \eta^t) + \frac{1}{2} \|\eta^t\|_2^2 \\ \text{s.t. } & \theta_d^t \geq 0, \forall d, t, \mathbf{z}_{dn}^t, \mathbf{y}_{dm}^t \geq 0, \forall d, n, m, t. \\ & \beta_k^t \in \mathcal{P}_U, \gamma_k^t \in \mathcal{P}_V, \forall k, t. \end{aligned} \quad (13)$$

where  $\mathcal{M}^t = \{\beta^t, \gamma^t\}$  is a set of parameters,  $f(\Theta^t, \mathcal{M}^t, D^t)$  is the objective function for unsupervised dSTC in time slice  $t$ , and  $R$  is the multi-class hinge loss:

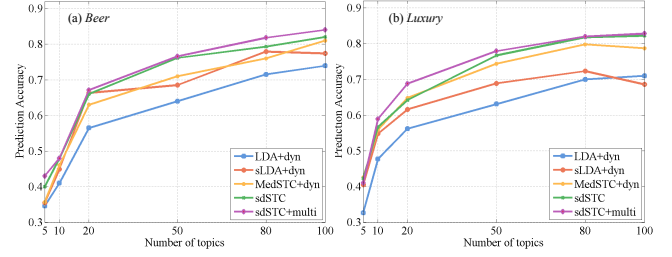
$$R(\Theta^t, \eta^t) = \frac{1}{|D^t|} \sum_{d=1}^{D^t} \max_y (\Delta(y_d, y) + \eta_y^\top \theta_d^t - \eta_{y_d}^\top \theta_d^t) \quad (14)$$

The above optimization problem can also be solved using the coordinated descent. Specifically, we alternate between

<sup>3</sup>Eq.(13) excludes brand competition terms such as  $\phi$  and  $\mathbf{r}$ .



**Figure 5: Comparison of accuracies of classification task (I-I) between our methods sdSTC and sdSTC+multi and the baselines of dLDA, sLDA, and MedSTC.**



**Figure 6: Comparison of accuracies of classification task (I-II) between our methods sdSTC and sdSTC+multi and the baselines of LDA+dyn, sLDA+dyn, and MedSTC+dyn.**

optimizing between  $\theta$  and  $\eta$ . It is worth noting that we learn different  $\eta^t$  for each time slice  $t$ .

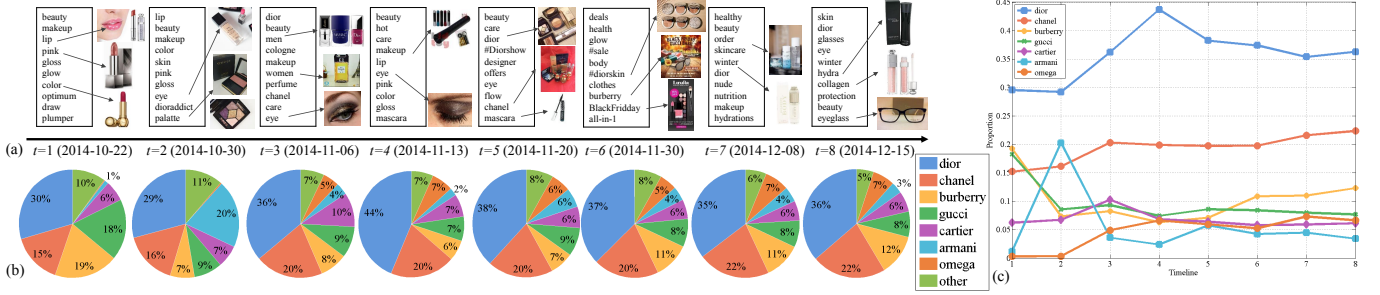
We design two experimental setups according to which data are used for training and test: (1) Task (I-I): we randomly divide the data in every time slice in  $[1, t]$  into two parts: 90% for training and 10% for test. (2) Task (I-II): we use the data in previous time slices  $[1, t-1]$  for training, and use all the data at time  $t$  for test.

For quantitative comparison, we run the following algorithms: (1) **sdSTC**: our model trained using text data from all time slice. (2) **sdSTC+multi**: our full model with multi-view extensions. (3) **dLDA** [5]: the dynamic LDA trained on all time slices, and then training a separate classifier for each time slice. (4) **MedSTC** [26]: the MedSTC trained using text data from all time slices. (5) **sLDA** [25]: the supervised LDA trained using text from all time slices. (6) **LDA+dyn** [3]: the LDA trained using text data from time slice  $t-1$ . (7) **sLDA+dyn** [25], the supervised LDA trained using text data from time slice  $t-1$ . (8) **MedSTC+dyn** [26]: the MedSTC trained using text data from time slice  $t-1$ . Note that the baselines of (3)–(5) are used for task (I-I), while the baselines of (6)–(8) are for task (I-II).

Since the **sLDA** and **sLDA+dyn** are too slow to learn on millions of documents, we randomly partition the *Beer* and *luxury* corpus into 10 and 15 groups, respectively, and then apply the algorithm into 5 randomly chosen groups, and report the average performance.

In the task (I-I), the training and test data lie in the same ranges of time slices. We compare our methods **sdSTC** and **sdSTC+multi** with the baselines of **dLDA**, **sLDA**, and **MedSTC**. We separately acquire the accuracy in every time slice, and then report the average accuracy. Fig.5 shows that our model outperforms all the other baselines for the two corpora. The accuracy increase of our method is more significant when the the number of topics is smaller. It is mainly because we add sparse terms on both document and word codes, leading to a less noisy document representation for a small number of topics. In addition, our **sdSTC+multi**





**Figure 4: The evolution of the topic *beauty* on the luxury corpus from 2014-10-22 to 2014-12-15. (a) Text and visual words associated with the topic on the timeline. (b) Evolution of brand competition pie graphs at every time slice. (c) Variation of proportions of competing brands over the topic.**

using both text and images achieves slightly better accuracies than our text-only *sdSTC*, which prove that text and images complement each other to detect better topics.

In the task (I-II), we compare our methods *sdSTC* and *sdSTC+multi* with the baselines of *LDA+dyn*, *sLDA+dyn*, and *MedSTC+dyn*. Different from the task (I-I), we train with the data up to time  $t-1$  and perform prediction for the data at  $t$ . Fig.6, show the results that our model achieves the best among all the methods, and the improvement of multi-view model over text-only model is significant, which indicates that image data is helpful to predict the *future*.

### 5.3.2 Temporal Localization

The second prediction task is, given an unseen past document  $d = (\mathbf{u}, \mathbf{v}, \mathbf{g})$ , to predict to which time slice it is likely to belong. This is closely related to the timestamp prediction in the research of social media diffusion. Specially, we solve the following problem in *dSTC*:

$$\max_t p(d|\mathcal{M}^t), \text{ where} \quad (15)$$

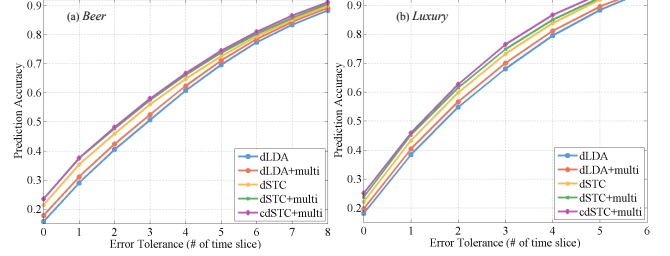
$$p(d|\mathcal{M}^t) = \prod_{n \in N_d} p(u_n|\beta^t) \prod_{m \in M_d} p(v_m|\gamma^t) \prod_{b \in B_d} p(g_b|\gamma^t)$$

is the likelihood of document  $d$  given the parameters in time slice  $t$ . Similar to the task (I-I) in the previous section, we randomly split the data of every time slice into 90% for training and 10% for localization test. We compare our methods *dSTC+multi* and *cdSTC+multi* (*i.e.* with or without brand competition-related terms) with the three baselines. (1) *dLDA+text*: dynamic LDA with only text data, (2) *dLDA+multi*: multi-view dynamic LDA using both text and image data, and (3) *dSTC+text*: *dSTC* with only text data.

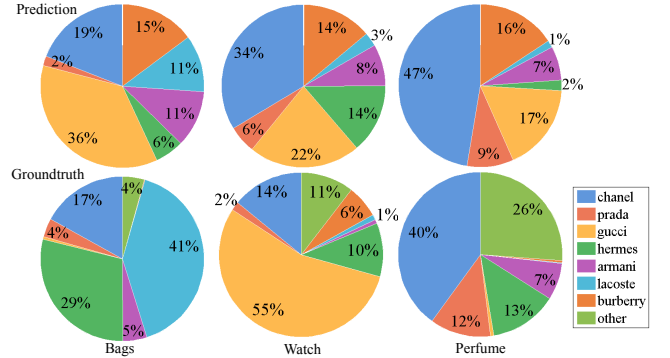
Fig.7 compares the average localization accuracies between our methods and baselines. We observe that our *dSTC* utilizing text, images and brands information achieve the best among all the methods. From a large accuracy rise from *dLDA+multi* to *cdSTC+multi*, we see that the explicit modeling of brand information helps improve the performance.

### 5.3.3 Prediction of Competition Trends

In the last prediction task, we use our model to capture the market competition dynamics on the timeline. We evolve the brand competition matrix  $\phi$  along with time, based on which we predict the future market competition trends using the past data. Specifically, we train our *cdSTC* model using the data in the range of  $[1, t-1]$ , and then predict the brand competition at  $t$ . Since there is no groundtruth for the brand occupation over the topics, we approximate the groundtruth as follows. We manually select the most interpretable top-



**Figure 7: Comparison of temporal localization accuracies between our methods *dSTC+multi* and *cdSTC+multi* and baselines *dLDA+text*, *dLDA+multi*, and *dSTC+text*.**

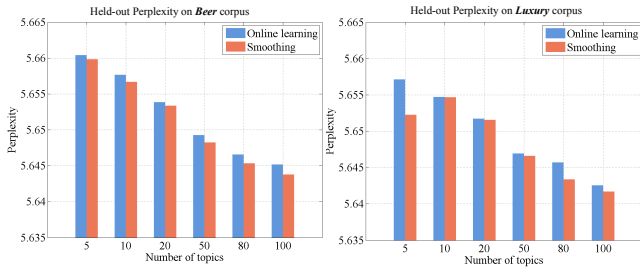


**Figure 8: The  $KL$ -divergence  $D(\text{prediction}||\text{groundtruth})$  are (bag, watch, perfume) = (0.4019 0.2615 0.0739).**

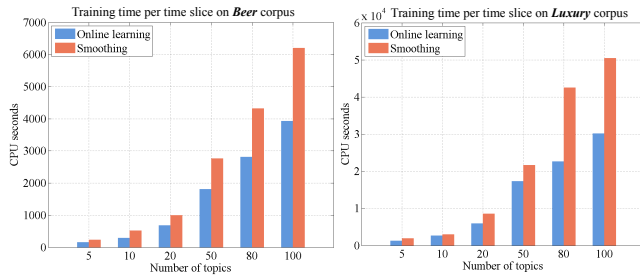
ics, such as *bag*, *watch*, and *perfume*. For each topic, we collect all tweets at time slice  $t$  that contain the topic word denoted by  $S$ . Then, for each brand, we count the tweets of  $S$  that include the brand name in the text. Finally, we build an  $L$ -dimensional normalized histogram, each bin of which implicitly indicates the proportion of the brand in the topic  $S$ . Fig.8 shows pie graphs comparing between the estimated  $\phi_k^t$  by our method and the approximated groundtruth. In the caption, we also report the  $KL$ -divergences for the three selected topics. Although it is hard to conclude that our prediction reflects well the actual proportions of brands over topics (mainly due to lack of accurate groundtruth), it is interesting to see that our method can visualize brand competitions over topics in a principled way while no previous method has addressed so far.

## 5.4 Online Learning and Smoothing

To provide a deep understanding of our model, we empirically compare between online learning and smoothing



**Figure 9: Held-out perplexity comparison between online learning and smoothing approach.**



**Figure 10: Training time comparison between online learning and smoothing approach.**

approach. We split 5% of data as a held-out test set, and train the models using the other data from all time slices, including 1.04 and 5.23 millions of tweets for *Beer* and *Luxury* corpora with associated images. Fig.9 shows the perplexity comparison between both approaches. We observe that online learning approach achieves a slightly higher perplexity than smoothing approach<sup>4</sup>, but both approaches does not show big difference with respect to the discovered topics and brand proportions. The training time of online learning approach is significantly shorter than that of smoothing approach, especially when the topic number is large. Therefore, online approach is more scalable on a large data set. Fig.10 shows the training time for both approaches. All experiments are performed in a single-thread manner on a desktop with Intel Core-I7 CPU and 32GB RAM.

## 6. CONCLUSION

We have presented a dynamic topic model for monitoring temporal evolution of market competition from a large collection of tweets and their associated images. Our model is designed to successfully address three major challenges: multi-view representation of text and images, competitiveness of multiple entities over shared topics, and tracking their temporal evolution. With experiments on a new twitter dataset consisting of about 10 millions of tweets and 8 millions of associated images, we showed that the proposed algorithm is more successful for the topic modeling and three prediction tasks over other candidate methods.

**Acknowledgement.** This work is supported by NSF Award IIS447676. The authors thank NVIDIA for GPU donations.

## 7. REFERENCES

[1] A. Ahmed and E. P. Xing. Timeline: A Dynamic Hierarchical Dirichlet Process Model for Recovering Birth/Death and Evolution of Topics in Text Stream. In *UAI*, 2010.

[2] N. Archak, A. Ghose, and P. G. Ipeirotis. Show me the Money! Deriving the Pricing Power of Product Features by Mining Consumer Reviews. In *KDD*, 2007.

[3] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet Allocation. *JMLR*, 3:993–1022, 2003.

[4] D. M. Blei and M. I. Jordan. Modeling Annotated Data. In *SIGIR*, 2003.

[5] D. M. Blei and J. D. Lafferty. Dynamic Topic Models. In *ICML*, 2006.

[6] J. Chang, J. L. Boyd-graber, S. Gerrish, C. Wang, and D. M. Blei. Reading Tea Leaves: How Humans Interpret Topic Models. In *NIPS*, 2009.

[7] N. Chen, J. Zhu, F. Sun, and X. Eric P. Large-Margin Predictive Latent Subspace Learning for Multiview Data Analysis. *IEEE PAMI*, 34:2365–2378, 2012.

[8] G. Doyle and C. Elkan. Financial Topic Models. In *NIPS Workshop for Applications for Topic Models: Text and Beyond*, 2009.

[9] Y. Feng and M. Lapata. Topic Models for Image Annotation and Text Illustration. In *NAACL HLT*, 2010.

[10] N. Glance, M. Hurst, K. Nigam, M. Siegler, R. Stockton, and T. Tomokiyo. Deriving Marketing Intelligence from Online Discussion. In *KDD*, 2005.

[11] T. Iwata, S. Watanabe, T. Yamada, and N. Ueda. Topic Tracking Model for Analyzing Consumer Purchase Behavior. In *IJCAI*, 2009.

[12] G. Kim, C. Faloutsos, and M. Hebert. Unsupervised Modeling and Recognition of Object Categories with Combination of Visual Contents and Geometric Similarity Links. In *ACM MIR*, 2008.

[13] T. Kurashima, T. Iwata, T. Hoshide, N. Takaya, and K. Fujimura. Geo Topic Model: Joint Modeling of User’s Activity Area and Interests for Location Recommendation. In *WSDM*, 2013.

[14] Q. Mei, X. Ling, M. Wondra, H. Su, and C. Zhai. Topic Sentiment Mixture: Modeling Facets and Opinions in Weblogs. In *WWW*, 2007.

[15] O. Netzer, R. Feldman, J. Goldenberg, and M. Fresko. Mine Your Own Business: Market-Structure Surveillance Through Text Mining. *Marketing Science*, 31(3):521–543, 2012.

[16] B. O’Connor, M. Krieger, and D. Ahn. TweetMotif: Exploratory Search and Topic Summarization for Twitter. In *ICWSM*, 2010.

[17] B. Pang and L. Lee. Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, 2:1–135, 2008.

[18] J. F. Prescott and S. H. Miller. *Proven Strategies in Competitive Intelligence: Lessons from the Trenches*. Wiley, 2001.

[19] I. Titov and R. McDonald. Modeling Online Reviews with Multi-grain Topic Models. In *WWW*, 2008.

[20] A. Vedaldi and K. Lenc. MatConvNet – Convolutional Neural Networks for MATLAB. In *CoRR*, 2014.

[21] Z. Wang, P. Cui, L. Xie, W. Zhu, Y. Rui, and S. Yang. Bilateral Correspondence Model for Words-and-Pictures Association in Multimedia-rich Microblogs. *ACM TOMM*, 10:2365–2378, 2014.

[22] S. Wu, W. M. Rand, and L. Raschid. Recommendations in Social Media for Brand Monitoring. In *RecSys*, 2011.

[23] P. Xie and E. P. Xing. Integrating Document Clustering and Topic Modeling. In *UAI*, 2013.

[24] K. Xu, S. S. Liao, J. Li, and Y. Song. Mining Comparative Opinions from Customer Reviews for Competitive Intelligence. *Decision Support Systems*, 50:743–754, 2011.

[25] J. Zhu, A. Ahmed, and E. P. Xing. MedLDA: Maximum Margin Supervised Topic Models. *JMLR*, 13:2237–2278, 2012.

[26] J. Zhu and E. P. Xing. Sparse Topical Coding. In *UAI*, 2011.

<sup>4</sup>A lower perplexity means a better generalization performance.