Topic Modeling in Large Scale Social Network Data

Aman Ahuja*, Wei Wei, Kathleen M. Carley

December 11, 2015
CMU-ISR-15-108

Institute for Software Research
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Center for Computational Analysis of Social and Organizational Systems:
CASOS technical report.

This work is an independent project that leverages the work that was in part supported by the Office of Naval Research N00014140737 with support from CMU for social media analytics by Dr. Carley and Mr. Wei. Additional support for this project was provided by the Center for Computational Analysis of Social and Organizational Systems (CASOS) at CMU. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Office of Naval Rsearch or the U.S. Government

*Undergraduate student, BITS Pilani – K.K. Birla Goa Campus, India

Keywords: Topic Modeling, Social Network Analysis, Probabalistic Graphical Models

Abstract

The growing popularity of social media such as Twitter and Facebook has made these websites an important source of information. The large amount of data available on these platforms presents new opportunities for mining information about the real world.

Because of its widespread usage, a lot of useful information can be extracted from the text available on these social media platforms. It can be used to infer important aspects about the users of these services and about the things happening in their surroundings.

This work proposes generative probabalistic models to identify latent topics and sentiments in social media data, mainly Twitter. In contrast to the majority of earlier work done in the field of topic modeling in social media data, this work incorporates various special characteristics of this data- mainly the short-length nature and special tokens like hashtags. The models proposed in work were compared qualitatively and quantitatively against several baseline models for evaluation. Experimental results suggest several improvements over the existing baseline techniques.

List of Figures

Contents

1. Introduction

The rapid growth of Internet in recent years has led to the growth of several social media websites like Twitter and Facebook in the recent years. People use these platforms to post about different aspects of their life and about the things happening in their surroundings. Using such platforms, people with similar interests can connect with each-other, create groups and share content such as messages, media with each other. Because of their increasing use and the vast quantity of data, this data can be used in several ways to gather information about the world, such as trending topics, breaking news and popular events.

In contrast to other forms of media such as newspaper, the text in the posts found on these websites is usually short in length, and concentrated on a much narrower selection of topics. Another interesting feature of social media data is the use of special tokens such as hashtags, that contain unique semantic meanings that are not captured by other ordinary words. Also, since a majority of people these days use handheld devices like mobile phones to access these services, a lot of data available on these platforms is geotagged. This information can be useful to determine various location-specific aspects around the world.

This thesis is focused on topic modeling as a means to discover latent topics in social media data, mainly Twitter. Several topic modeling techniques have been proposed in the recent years. Most of these models are based on the Latent Dirichlet Allocation [1]. But whether these techniques can be used to model social media text, which differs from other forms of text in variety of ways has not been well studied.

In this work, we address the challenge of modeling social media text using Bayesian graphical models that take into account the special characteristics of the social media text, such as their short-length nature and special tokens such as hashtags. We also present both qualitative and quantitative evaluation of the proposed models against several baseline models. The subsequent chapters are organized as follows:

- **Chapter 2** gives an overview of the several topic modeling techniques that have been proposed so far.
- **Chapter 3** describes the Twitter dataset that was used to evaluate the models presented in this work
- **Chapter 4** presents a generative model, namely SMTM(*Social Media Topic Model*) to discover latent topics in social media data. This model characterizes both words and hashtags separately, and takes into account the short-length nature of social media posts.
- **Chapter 5** presents a sentiment topic model, namely SMSTM(*Social Media Sentiment Topic Model*). This model is an extension of SMTM, but also incorporates the sentiment.
- **Chapter 6** outlines the major contributions of the work presented in this work, which is followed by the outline directions for future work.

2. Related Work

This chapter presents an overview of the several previous works that are related to this thesis. The focus here will be on 3 main categories- topic modeling, sentiment analysis and modeling social media data.

2.1. Topic Modeling

The success of topic modeling in recent years has gained a lot of interest among the research community. A topic model is a probabalistic model that can be used to discover latent topics in a corpus of documents. One of the earliest technique in the field of topic modeling was the probabalistic Latent Semantic Indexing (pLSI) proposed by Hoffman [2]that models a document as a mixture of topics. pLSI models each document as a mixture over topics, but there is no generative process for determining the document-topic distribution, which leads to problems while assigning probabilities to documents outside the training set. Most of the recent research in the field of topic modeling is based on the Latent Dirichlet Allocation [1] proposed by Blei. LDA overcomes the shortcomings of pLSI by modeling each document as a mixture over topics, and each topic as a mixture over words.

2.2. Sentiment Analysis

Sentiment analysis of social media data remains a key area of research. A lot of techniques ([3], [4]) have been proposed to detect sentiment polarity of Twitter messages. A majority of work in the field of sentiment analysis for Twitter data aims to classify the polarity of individual messages, and not of the topics as a whole. Unlike these work, we focus on learning the latent representations of the sentiment topic as well as the documents instead of predicting the sentiment label of the individual messages.

One of the earliest work that incorporates sentiment associated with topics using a generative model model was the Joint Sentiment Topic(JST) model [5]. JST models each each document as a mixture over topics and sentiments. The prior sentiment knowledge about different words is used in the initialization step while assigning polarity to different words in each document. In this way, JST models the documents as a mixture of positive and negative topics. More recently, [6] proposed ASUM, that assigns topic and sentiment at the sentence level, unlike JST that assigns topic and sentiment at the word level. But since ASUM generates topics from sentiment, it finds senti-aspects, and does not perform reasonably well to find positive and negative aspects of each topic. Also, when applied to social media data, both JST and ASUM do not treat words and hashtags separately.

2.3. Modeling Social Media Data

A number of techniques based on LDA have been proposed for social media data. The Author-Topic model proposed in [7] that can be used to determine the topic distributions of various authors in the dataset. [8] discussed the application of the this model to Twitter data. But this model generally does not fit well in case of social media data where the documents are usually short in length, and belong to a single topic. [9] takes into account this property, and proposed Twitter-LDA model, that assigns topics at the tweet level, but does not treat both words and hashtags separately. Apart from the growing usage of topic modeling techniques for text, some of the recent work also aims to to use these techniques for other forms of data, such as the network dataset in social networks. The SSN-LDA model [10] is one such work that tries to model communities in social networks using a generative model.

The SMTM and SMSTM models proposed in this work are largely inspired by Twitter-LDA model and ASUM on the fact that topics are assigned at the document level. In addition, SMTM and SMSTM treat both words and hashtags separately. Also, SMSTM aims to find the positive and negative aspects of each topic, unlike ASUM, that discovers positive and negative topics.

3. Dataset

This chapter gives details about the dataset used to evaluate the models presented in this work.

3.1. Twitter Dataset

To evaluate the models, I used Twitter dataset collected using the Twitter Streaming API [1]. When collecting the data, the geo-region bounding that was selected roughly covered the entire area of USA. This dataset was then preprocessed before it could be used for the models described in this thesis. In total, there were around 2.4 million tweets collected within a 30-day time period from May 1, 2011 to May 31, 2011.

Table 3.1: Dataset Statistics

Number of users(U)	11509
Number or unique words(W)	557318
Number of unique hashtags(H)	100445

3.1.1. Special characteristics of Twitter ”tweets”

In contrast to other forms of text, the text in tweets is relatively short in length, restricted by the limit on the number of characters, which is *140* in case of Twitter. Because of this, the text also contains a lot of abbreviations, so that the information can be conveyed with limited number of characters. It is observed that tweets generally contain a lot of mis-spelled words also. This makes topic mining and text analysis using Twitter data a challenging task.

Hashtags: A hashtag is a meta tag frequently used in social media posts, that can be used to link the post to a specific theme or topic. It is generally observed that popular events and topics are characterized by common hashtags, and it makes it easier to find the posts related to that topic. For example, people might use the tag *#Halloween*, if they tweet about something that is related to Halloween festival.

3.2. Preprocessing

All the tweets used to evaluate the models were first preprocessed to remove the noisy and irrelevant words. The various steps involved in preprocessing stage were as follows:

- **Tokenization of emoticons:** Since emoticons are useful in sentiment analysis, the first step was to replace all the valid emoticons with different tokens, so that they were not lost while removing the punctuation marks.

[1] https://dev.twitter.com/streaming/overview

- **Conversion to lowercase:** All the letters in the dataset were converted to lowercase in order to prevent duplicates, and preserve the semantic meaning of same words that had different case letters.

- **URL and co-mentions removal:** The third step was the removal of URLs and co-mentions, so that the text contains only meaningful words.

- **Stop word removal:** Since stop words like *for, the* do not convey any meaning and are not topic-specific words, these words were also removed.

- **Removing infrequent words:** Since words that occur very frequently in the corpus (less than 2 times) are more likely to be mis-spelled words, all such words from the dataset.

- **Restoring emoticons and tokenization:** The final step of the preprocessing stage involved replacing the emoticon tokens assigned in step-1 with the original emoticons. This was followed by tokenization of all the words and tags. Tokenization of all the tweets in preprocessing stage improves performance, since we do not need to tokenize the entire corpus during run-time.

4. SMTM: Social Media Topic Model

Given the growing usage of social media services, it has become increasingly important to determine what are the key topics that are dominant on these platforms. This can give an insight about the major things happening around in the world such as major events, disasters, etc.

This chapter proposes SMTM(*Social Media Topic Model*)- a probabalistic model to discover latent topics in social media data. In contrast to other previously defined models, SMTM takes into account special characteristics of social media data, which distinguishes it from other models.

4.1. Model Description

SMTM models the generative process of social media posts that contain both words and hashtags. In contrast to LDA, SMTM treats both words and hashtags separately and gives a topic-word distribution ϕ and topic-hashtags distribution η for each topic. Also, since social media posts are generally short in length (*eg., 140 characters in Twitter*), it is highly likely that all words in a tweet belong to the same topic. SMTM takes into account this assumption, and assigns topic at the document level for each social media post. It models each user u as a mixture over topics (*or interests*), and then generates the topic z for each post by the user based on the user-topic distribution θ_u. It then assigns this topic to all the words and hashtags in the post.

It is also observed that some topics (*eg., those related to a popular event*) contain a higher proportion of hashtags than other topics. SMTM also incorporates this fact using a dependency from the topic z to the category of the word token c. The value of this category variable c determines whether a token is a word or a #tag.

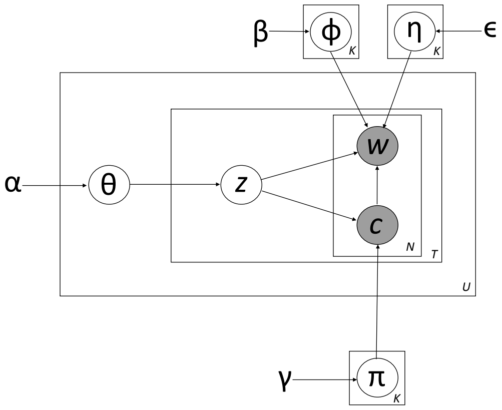

Figure 4.1: Plate notation of SMTM

4.2. Generative Process

The overall generative process of SMTM can be described as follows:

- For each topic k,
  - Draw topic-word distribution $\phi_k \sim Dirichlet(\beta)$
  - Draw topic-tag distribution $\eta_k \sim Dirichlet(\epsilon)$
  - Draw topic-category distribution $\pi_k \sim Dirichlet(\gamma)$
- For each user u, draw user-topic distribution $\theta_u \sim Dirichlet(\alpha)$
- For each post t by user u, choose a topic $z_{ut} \sim Multinomial(\theta_u)$
- For each token n in the post t by user u,
  - Choose a category $c_{utn} \sim Bernoulli(\pi_{z_{ut}})$
  - Draw a word/tag w_{utn} as follows:

$$w_{utn} \sim \begin{cases} Multinomial(\phi_{z_{ut}}), & \text{if } c_{utn} = 1 \\ Multinomial(\eta_{z_{ut}}), & \text{if } c_{utn} = 0 \end{cases}$$

4.3. Inference

The joint probability distribution of SMTM can be given by he following equation:

$$\begin{aligned} P(\boldsymbol{Z}, \boldsymbol{W}, \boldsymbol{C}, &\theta, \phi, \eta, \pi | \alpha, \beta, \epsilon, \gamma) \\ &= \prod_{i_1=1}^{K} P(\phi_{i_1}|\beta) \prod_{i_2=1}^{K} P(\eta_{i_2}|\epsilon) \prod_{i_3=1}^{K} P(\pi_{i_3}|\gamma) \prod_{u=1}^{U} P(\theta_u|\alpha) \\ &\prod_{t=1}^{T} P(Z_{ut}|\theta_u) \prod_{n=1}^{N} P(C_{utn}|\pi_{Z_{ut}}) P(W_{utn}|C_{utn}, \phi_{Z_{utn}}, \eta_{Z_{utn}}) \end{aligned} \tag{4.1}$$

To infer the latent variable z, we use the collapsed Gibbs sampling technique described in [11]. The model parameters θ, π, ϕ and η were first integrated out, which gives the following distribution:

$$\begin{aligned} P(\boldsymbol{Z}, \boldsymbol{W}, \boldsymbol{C} | \alpha, \beta, \epsilon, \gamma) = \\ &\prod_{u=1}^{U} \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \frac{\prod_{i=1}^{K} \Gamma(N_u^i + \alpha_i)}{\Gamma(\sum_{i=1}^{K} N_u^i + \alpha_i)} \\ &\prod_{i_1=1}^{K} \frac{\Gamma(\sum_{r=1}^{W} \beta_r)}{\prod_{r=1}^{W} \Gamma(\beta_r)} \frac{\prod_{r=1}^{W} \Gamma(M_{w_r}^{i_1} + \beta_r)}{\Gamma(\sum_{r=1}^{W} M_{w_r}^{i_1} + \beta_r)} \\ &\prod_{i_2=1}^{K} \frac{\Gamma(\sum_{r=1}^{H} \epsilon_r)}{\prod_{r=1}^{H} \Gamma(\epsilon_r)} \frac{\prod_{r=1}^{H} \Gamma(M_{h_r}^{i_2} + \epsilon_r)}{\Gamma(\sum_{r=1}^{H} M_{h_r}^{i_2} + \epsilon_r)} \\ &\prod_{i_3=1}^{K} \frac{\Gamma(\sum_{r=0}^{1} \gamma_r)}{\prod_{r=0}^{1} \Gamma(\gamma_r)} \frac{\prod_{r=0}^{1} \Gamma(C_r^i + \gamma_r)}{\Gamma(\sum_{r=0}^{1} C_r^i + \gamma_r)} \end{aligned} \tag{4.2}$$

The only variables left after integration are z, w and c. Since w and c are observed variables, we only sample z for each post (u, t) since it is the only latent variable left after integration. It is done according to the following equation:

$$\begin{aligned} P(z_{ut} = k|\boldsymbol{Z_{-ut}}, \boldsymbol{C}, \boldsymbol{W}, \alpha, \beta, \gamma, \epsilon) \propto & \frac{N_u^{k,-ut} + \alpha_k}{\sum_{i=1}^{K} N_u^{i,-ut} + \alpha_i} \\ & \frac{\prod_{r \in W_{ut}} \prod_{j=0}^{n_{ut}^{w,r}-1} (M_{w_r}^{k,-ut} + \beta_r + j)}{\prod_{j=0}^{n_{ut}^{w,(.)}-1} ((\sum_{r=1}^{W} M_{w_r}^{k,-ut} + \beta_r) + j)} \\ & \frac{\prod_{r \in H_{ut}} \prod_{j=0}^{n_{ut}^{h,r}-1} (M_{h_r}^{k,-ut} + \epsilon_r + j)}{\prod_{j=0}^{n_{ut}^{h,(.)}-1} ((\sum_{r=1}^{H} M_{h_r}^{k,-ut} + \epsilon_r) + j)} \\ & \frac{\prod_{r=0}^{1} \prod_{j=0}^{n_{ut}^{r,(.)}-1} (C_r^{k,-ut} + \gamma_r + j)}{\prod_{j=0}^{n_{ut}^{(.),(.)}-1} ((\sum_{r=0}^{1} C_r^{k,-ut} + \gamma_r) + j)} \end{aligned} \tag{4.3}$$

After sampling, the model parameters can be recovered using the following equations:

$$\theta_u^k = \frac{N_{u,(.)}^k + \alpha_k}{\sum_{i=1}^{K} N_{u,(.)}^i + \alpha_i} \tag{4.4}$$

$$\phi_k^r = \frac{M_{w_r}^k + \beta_r}{\sum_{r=1}^{W} M_{w_r}^k + \beta_r} \tag{4.5}$$

$$\eta_k^r = \frac{M_{h_r}^k + \epsilon_r}{\sum_{r=1}^{H} M_{h_r}^k + \epsilon_r} \tag{4.6}$$

$$\pi_k^c = \frac{C_c^k + \gamma_c}{\sum_{r=0}^{1} C_r^k + \gamma_r} \tag{4.7}$$

The definitions of all the equations is given in Table 4.1 and Table 4.2.

U	the number of users
T	the number of posts/tweets
N	the number of tokens(words and hashtags) in each post
K	the number of topics
W	the size of word vocabulary
H	the size of hashtag vocabulary
z	topic
w	word
c	category (word or hashtag)
θ	user-topic distribution
ϕ	topic-word distribution
η	topic-hashtag distribution
π	topic-token category distribution
α	Dirichlet prior vector for θ
β	Dirichlet prior vector for ϕ
ϵ	Dirichlet prior vector for η
γ	Dirichlet prior vector for π, π_s
λ	Dirichlet prior vector for ψ

Table 4.1: Notations: SMTM

N_u^k	number of tweets by user u that occurred in topic k
$M_{w_r}^k$	number of occurrences of r^{th} word from word vocabulary in topic k
$M_{h_r}^k$	number of occurrences of r^{th} hashtag from hashtag vocabulary in topic k
C_r^k	number of occurrences of tokens from category r in topic k
W_{ut}	set of unique words in the post *(u,t)*
H_{ut}	set of unique hashtags in the post *(u,t)*
$n_{u,t}^{x,r}$	number of occurrences of r^{th} token from vocabulary x in post *(u,t)*

Table 4.2: Auxiliary Notations: SMTM

[1]

4.4. Experimental Results

4.4.1. Experimental Setup

In order to evaluate SMTM, we first need to input the values of the hyperparameters α, β, γ and ϵ. These hyperparameters serve as a priori for the model. We used symmetric values for all the hyperparameters, which were derived experimentally. Specifically, we set $\alpha = 1$, $\beta = 0.05$, $\epsilon = 0.05$ and $\gamma = 5$. The model was run for 800 iterations, using different values for the number of topics, K.

4.4.2. Qualitative Results

In order to demonstrate the qualitative results, two topics from the results were selected and their topi 10 words and hashtags were picked based on the corresponding values of the topic-word distribution ϕ and topic-tag distribution η. These results are presented in Table 4.3.

As it is evident from the results shown in Table 4.3, first topic contains words and tags that are related to a particular event, i.e., the death of Osama Bin Laden, since the Twitter dataset was from May, 2011 (the same time when US assassinated Osama Bin Laden). The second topic mostly has words related to food, particularly good food as it contains words like "eat", "good", "food", etc. These words are supported by corresponding hashtags like "#fattweet", "#yum", "#hungry", etc.

Topic-category distribution: We compare the value of the parameter π for different topics and examine the corresponding words and hashtags for each topic. It is observed that for a majority of topics, the ratio $\pi_{k,0} : \pi_{k,1}$ of number of words to the number of hashtags assigned to that topic is around *0.25*. Some topics have a high distribution of hashtags as compared to other topics. After examining the corresponding words and hashtags for these topics, it was observed that most of the topics with higher proportion

[1] For all the terms shown in the equations,

- (-u,t) denotes that the term excludes the current post (u,t)
- for any dimension d, (.) denotes that the term is not limited to the specific value of d

of hashtags were associated with advertising campaigns or related to news. Figure 4.2 shows the values of $\pi_{k,0}(\#tags)$ and $\pi_{k,1}(words)$ for all the topics when $K = 60$.

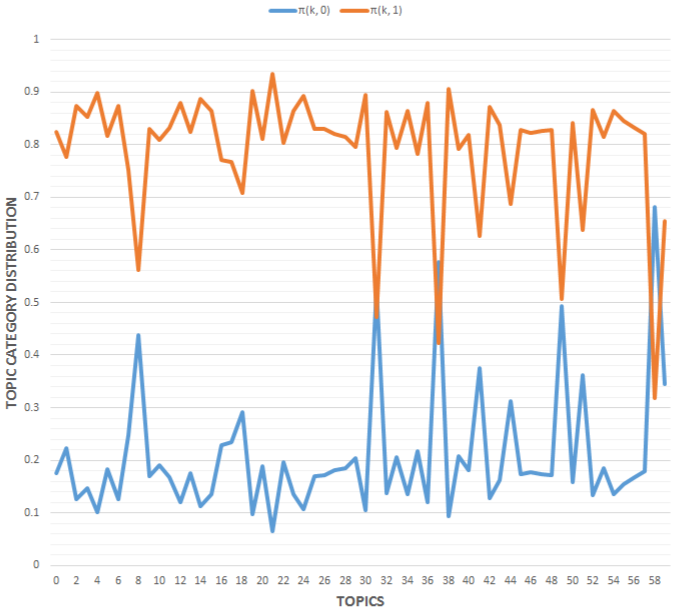

Figure 4.2: Topic-category distribution with K = 60

4.4.3. Quantitative Results

To compare SMTM quantitatively with other models, we choose LDA as the baseline model and compare the perplexity of both the models, which is a commonly used criterion for evaluating topic models. The perplexity of a model for a test set containing M documents is defined as:

$$Perp(D_{test}) = exp\left\{ \frac{-\sum_{d=1}^{M} log\ p(w_d)}{\sum_{d=1}^{M} N_d} \right\} \tag{4.8}$$

Since we are interested in comparing the perplexity of SMTM with LDA, the exponent term can be ignored. Th perplexity of SMTM can be calculated as per the following

T1:Words	T1:#tags	T2:Words	T2:#tags
bin	#caseyanthonytrial	eat	#fattweet
laden	#osama	good	#win
obama	#syria	food	#yum
osama	#news	chicken	#yummy
news	#obama	:)	#hungrytweet
dead	#pakistan	icecream	#hungry
death	#binladen	eating	#munchies
world	#usa	breakfast	#love
killed	#osamabinladen	cheese	#delicious
man	#dead	drink	#ny

Table 4.3: Sample words and hashtags for 2 different topics obtained using SMTM

equation:

$$Perp(D_{test}^{SMTM}) = \frac{1}{\sum_{u=1}^{U}\sum_{t=1}^{T} N_{ut}} \sum_{u=1}^{U}\sum_{t=1}^{T} log(\sum_{k=1}^{K} \theta_{u,k}(\sum_{n=1}^{N_w^{ut}} \pi_{k,1}\phi_{k,n} + \sum_{n=1}^{N_h^{ut}} \pi_{k,0}\eta_{k,n})) \tag{4.9}$$

As described in [1], a lower perplexity score indicates better predictive performance of the model. A high likelihood value indicates that model has a better predictive accuracy. Since perplexity is the negative log of the likelihood *p(w)*, a model with lower perplexity is more likely to have a better predictive performance.

The perplexity of SMTM was compared with that of LDA, using different values of K ranging from 5 to 100. The perplexity comparison is shown in Figure 4.3. The lower perplexity of SMTM against LDA indicates that SMTM has a better predictive performance in case of social media data.

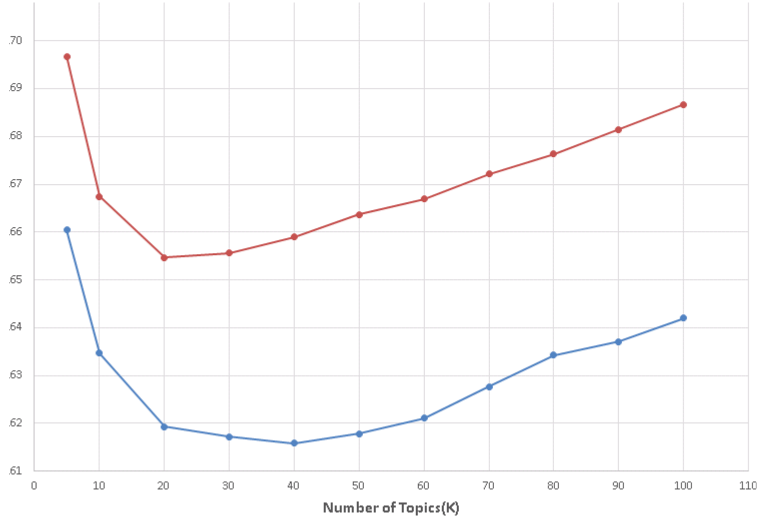

Figure 4.3: Perplexity comparison of SMTM with LDA

4.4.4. Running time

We now show the running time per Gibbs sampling iteration for the corpus containing 2.38 million tweets. It is observed that the running time increases almost linearly as the number of topics K increases. This is shown in Figure 4.4. This is because as the number of topics increases, for each post $(u,\ t)$, the number of times that we need to calculate the marginal probability of latent variables also increases.

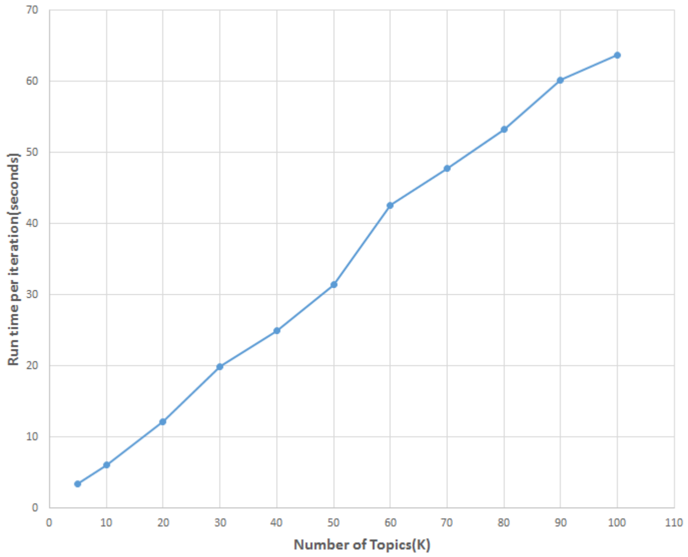

Figure 4.4: Running time per iteration for SMTM

4.5. Conclusion

In this chapter, we presented a novel topic model to discover latent topics in social media dataset. One key characteristic of this model was that it is particularly designed for social media text, which differs from other forms of text in a variety of ways. We evaluated our model on Twitter dataset, although since the structure of data on different social meda platforms is similar, we believe that the model can perform reasonably well on other datasets also. We compared our model with the existing baseline model and found that it outperforms the baseline model.

5. SMSTM: Social Media Sentiment Topic Model

Chapter 4 introduced a novel method to discover latent topics from social media data. In addition to discovering topics, it is equally important to determine the sentiments associated with the topics. It can be useful in determining whether a topic is good or bad, based on the sentiment polarity associated with topic. For example, a topic associated with a natural disaster like tornado has negative sentiment, but a topic that describes nightlife and holidays has positive polarity. Also, there are some topics that have both positive and negative aspects. For example, topic associated with Presidential elections in the United States can have both positive and negative aspects associated with different candidates contesting for he elections. To tackle this problem, we introduce SMSTM(*Social Media Sentiment Topic Model*), that can discover topics and their sentiment from a corpus containing social media data.

5.1. Model Description

SMSTM is a generative model that can discover latent topics and sentiments in social media data. This model is an extension of SMTM, but it also incorporates the sentiment associated with the topics. The graphical model for SMSTM is shown in Figure 5.1.

In addition to all the other variables in SMTM, SMSTM has a sentiment variable s at the document level, which is the sentiment polarity of the document. This is drawn from the sentiment distribution ψ_z of the topic z associated with the document, which can determine the sentiment associated the topic. For each token in the document (u, t), after determining the category (word or hashtag) of the token, it is drawn from the respective topic-sentiment-word distribution $\phi_{k,s}$ or $\eta_{k,s}$ based on the value of the variable c. The prior sentiment polarity of words can be incorporated into SMSTM in the values of the hyperparameters β and ϵ based on the assumption that since a word with positive sentiment polarity is more likely to be in a positive sentiment topic.

Intuitively, the model can be described as follows: whenever a user u, decides to write a post t, he first decides the topic z_{ut} of the post based on his interest distribution θ_u. He then decides the sentiment s_{ut} and the type(word or hashtag) of the tokens in the post. Finally, he generates the tokens w_{utn} based on the topic, sentiment and category of the tokens.

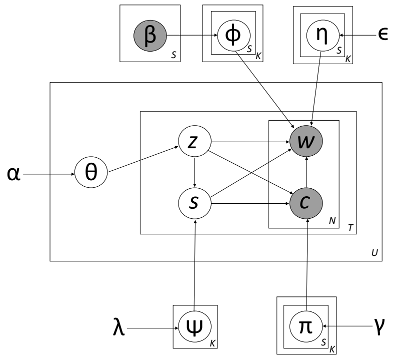

Figure 5.1: Plate notation of SMSTM

5.2. Generative Process

The generative process of SMSTM can be described as follows:

- For each topic k,
  - Draw topic-sentiment distribution $\psi_k \sim Dirichlet(\lambda)$
  - For each sentiment s,
    - Draw topic-sentiment-category distribution $\pi_{k,s} \sim Dirichlet(\gamma)$
    - Draw topic-sentiment-word distribution $\phi_{k,s} \sim Dirichlet(\beta_s)$
    - Draw topic-sentiment-hashtag distribution $\eta_{k,s} \sim Dirichlet(\epsilon_s)$
- For each user u,
  - Draw user-topic distribution $\theta_u \sim Dirichlet(\alpha)$
  - For each post t by the user,
    - Choose a topic $z_{ut} \sim Multinomial(\theta_u)$
    - Choose a sentiment $s_{ut} \sim Multinomial(\psi_{z_{ut}})$
    - For each token n in the post $(u,\ t)$,
      - Choose a category $c_{utn} \sim Multinomial(\pi_{z_{ut},s_{ut}})$
      - Draw a word/hashtag as follows:

$$w_{utn} \sim \begin{cases} Multinomial(\phi_{z_{ut},s_{ut}}), & \text{if } c_{utn} = 1 \\ Multinomial(\eta_{z_{ut},s_{ut}}), & \text{if } c_{utn} = 0 \end{cases}$$

5.3. Inference

The joint probability distribution for SMSTM can be given as:

$$\begin{aligned} & P(\boldsymbol{Z}, \boldsymbol{S}, \boldsymbol{W}, \boldsymbol{C}, \theta, \psi, \pi, \phi, \eta | \alpha, \beta, \epsilon, \gamma, \lambda) \\ & \qquad = \prod_{i_3=1}^{K} \prod_{s=1}^{S} P(\pi_{i_3,s}|\gamma_s) \prod_{i_4=1}^{K} P(\psi_{i_4}|\lambda) \\ & \qquad \prod_{i_1=1}^{K} \prod_{s=1}^{S} P(\phi_{i_1,s}|\beta_s) \prod_{i_2=1}^{K} \prod_{s=1}^{S} P(\eta_{i_2,s}|\epsilon_s) \\ & \qquad \prod_{u=1}^{U} P(\theta_u|\alpha) \prod_{t=1}^{T} P(z_{ut}|\theta_u) P(s_{ut}|\psi_{z_{ut}}) \\ & \qquad \prod_{n=1}^{N} P(C_{utn}|\pi_{z_{ut},s_{ut}}) P(W_{utn}|C_{utn}, \phi_{z_{ut},s_{ut}}, \eta_{z_{ut},s_{ut}}) \end{aligned} \tag{5.1}$$

Similar to SMTM, the inference in SMSTM is also done using collapsed Gibbs sampling. All the model parameters θ, ψ, ϕ, η and π are integrated out easily because of the Dirichlet-Multinomial conjugacy. In addition to the topic variable z, SMSTM has one additional latent variable s that needs to be sampled for each tweet $(u,\ t)$. For each post $(u,\ t)$, this sampling can be done as per the following equation:

$$\begin{gathered}
P(z_{ut} = k, s_{ut} = p | \boldsymbol{Z_{-ut}}, \boldsymbol{S_{-ut}}, \boldsymbol{C}, \boldsymbol{W}, \alpha, \beta, \epsilon, \gamma, \lambda) \propto \\
\frac{N_{u,(.)}^{k,-ut} + \alpha_k}{\sum_{i=1}^{K} N_{u,(.)}^{i,-ut} + \alpha_i} \frac{L^{k,p,-ut} + \lambda_p}{\sum_{s=0}^{1} L^{k,s,-ut} + \lambda_s} \\
\frac{\prod_{r \in W_{ut}} \prod_{j=0}^{n_{ut}^{w,r}-1} (M_{w_r}^{k,p,-ut} + \beta_r + j)}{\prod_{j=0}^{n_{ut}^{w,(.)}-1} ((\sum_{r=1}^{W} M_{w_r}^{k,p,-ut} + \beta_r) + j)} \\
\frac{\prod_{r \in H_{ut}} \prod_{j=0}^{n_{ut}^{h,r}-1} (M_{h_r}^{k,p,-ut} + \epsilon_r + j)}{\prod_{j=0}^{n_{ut}^{h,(.)}-1} ((\sum_{r=1}^{H} M_{h_r}^{k,p,-ut} + \epsilon_r) + j)} \\
\frac{\prod_{r=0}^{1} \prod_{j=0}^{n_{ut}^{r,(.)}-1} (C_r^{k,p,-ut} + \gamma_r + j)}{\prod_{j=0}^{n_{ut}^{(.),(.)}-1} ((\sum_{r=0}^{1} C_r^{k,p,-ut} + \gamma_r) + j)}
\end{gathered} \tag{5.2}$$

The model parameters θ, ψ, ϕ, η and π can then be calculated as per the following equations:

$$\theta_u^k = \frac{N_{u,(.)}^{k} + \alpha_k}{\sum_{i=1}^{K} N_{u,(.)}^{i} + \alpha_i} \tag{5.3}$$

$$\phi_{k,p}^r = \frac{M_{w_r}^{k,p} + \beta_{p,r}}{\sum_{r=1}^{W} M_{w_r}^{k,p} + \beta_{p,r}} \tag{5.4}$$

$$\eta_{k,p}^r = \frac{M_{h_r}^{k,p} + \epsilon_{p,r}}{\sum_{r=1}^{H} M_{h_r}^{k,p} + \epsilon_{p,r}} \tag{5.5}$$

$$\pi_{k,p}^c = \frac{C_c^{k,p} + \gamma_c}{\sum_{r=0}^{1} C_r^{k,p} + \gamma_r} \tag{5.6}$$

$$\psi_k^p = \frac{L_{(.)}^{k,p} + \lambda_p}{\sum_{s=0}^{1} L_{(.)}^{k,s} + \lambda_s} \tag{5.7}$$

(All the notations are described in Table 5.1 and Table 5.2)

5.4. Sentiment Lexicon

To incorporate the prior sentiment polarity of words in SMSTM, Vader sentiment lexicon[12] was used. This choice was made based on the fact that Vader is specifically designed for words that frequently occur in social media posts, particularly Twitter and is highly optimized for such datasets. Also, a lot of these commonly occurring polar words are present only in Vader, and cannot be found in other sentiment lexicons like the MPQA subjectivity corpus[14] and SentiWordnet[13]. Since in our experiments, we consider only positive and negative sentiments, we separate out the positive and negative sentiment words from Vader based on their score. After this, the sentiment lexicon had 3300 positive sentiment words and 4100 negative sentiment words.

U	the number of users
T	the number of posts/tweets
N	the number of tokens(words and hashtags) in each post
K	the number of topics
S	the number of sentiments
W	the size of word vocabulary
H	the size of hashtag vocabulary
z	topic
w	word
c	category (word or hashtag)
s	sentiment
θ	user-topic distribution
ϕ	topic-word distribution
η	topic-hashtag distribution
π	topic-token category distribution
ψ	topic-sentiment distribution
α	Dirichlet prior vector for θ
β_s	Dirichlet prior vector for ϕ_s
ϵ_s	Dircihlet prior vector for η_s
γ_s	Dirichlet prior vector for π_s
λ	Dirichlet prior vector for ψ

Table 5.1: Notations: SMSTM

$N^k_{u,t}$	number of times tweet *(u, t)* has occurred in topic k
W_{ut}	set of unique words in the post *(u,t)*
H_{ut}	set of unique hashtags in the post *(u,t)*
$n^{x,r}_{u,t}$	number of occurrences of r^{th} token from vocabulary x in post *(u,t)*
$M^{k,p}_{w_r}$	number of occurrences of r^{th} word from word vocabulary in topic k with polarity p
$M^{k,p}_{h_r}$	number of occurrences of r^{th} hashtag from hashtag vocabulary in topic k with polarity p
$C^{k,p}_r$	number of occurrences of tokens from category r in topic k with polarity p
$L^{k,p}$	total number of posts that are assigned topic k and p

Table 5.2: Auxiliary Notations

5.5. Experimental Results

5.5.1. Experimental Setup

To evaluate SMSTM, the same Twitter dataset as the one used for SMTM was used (2.4 million tweets). The number of sentiments (S) was set to 2, since we were only interested in positive and negative topics. The hyperparameters α, λ and γ were assigned symmetric values, which were determined experimentally. These were $\alpha = 1$, $\lambda = 5$ and $\gamma = 5$. As described earlier, the prior sentiment knowledge in SMSTM is incorporated by making β and ϵ unsymmetrical vectors.

Since hashtags are not proper words that can be found in the english vocabulary, the hyperparameter ϵ was assigned symmetric value equal to 0.05. For each word r that was present in the sentiment lexicon, the value of β was assigned as follows:

$$\beta_{r_s} = \begin{cases} 0.09, & \text{if polarity(r)=s} \\ 0.01, & \text{if polarity(r)} \neq s \end{cases}$$

For all the other words r whose prior sentiment knowledge was not known, a symmetric β_r was assigned which was equal to 0.05.

During the initialization step for each post $(u,\ t)$, the number of positive words(pos) and negative words(neg) was calculated by comparing each word in post $(u,\ t)$ against the sentiment lexicon. After this, the sentiment s_{ut} was assigned as follows:

$$s_{ut} = \begin{cases} 1, & \text{if pos>neg} \\ 0, & \text{if pos<neg} \\ random\{0,1\}, & \text{otherwise} \end{cases}$$

The model was run for 800 Gibbs sampling iterations with different values of K, ranging from 5 to 100.

5.5.2. Qualitative Results

This section presents the words and hashtags obtained for different topic, and gives an overview of how to determine the topic polarity using SMSTM. In SMSTM, we use the value of the value of the parameter ψ_k to determine the polarity of the topic k. This sentiment polarity can be verified by examining the set of sentiment words obtained for each topic. This is illustrated in Table 5.3.

As it is evident from the words shown in Table 5.3, the topic shown here is about music and awards, since it contains tokens like ”music”, ”video” and ”billboardawards”. SMSTM gives a set of both positive and negative words and #tags associated with this topic. The value of $\psi_{k,1}$ for this topic is much greater than the value of $\psi_{k,0}$ which indicates that this topic is more likely to be a positive topic.

T1: +ve Words	**T1: +ve #tags**	**T1: -ve Words**	**T1: -ve #tags**
lol	#billboardawards	lol	#lmao
love	#thevoice	whoa	#billboards
song	#americanidol	lil	#loud
beyonce	#idol	shit	#garbage
video	#nowplaying	voice	#np
gaga	#1	video	#co
music	#beyonce	online	#boaw
sing	#oprah	internet	#justsaying
good	#winning	song	#fb
performance	#teamminaj	watch	#bored
$\boldsymbol{\psi_{k,1}} = \mathbf{0.9722124516355962}$		$\boldsymbol{\psi_{k,0}} = \mathbf{0.027787548364403798}$	

Table 5.3: Sample positive and negative words and hashtags for a topic obtained using SMSTM

5.5.3. Quantitative Results

To evaluate SMSTM quantitatively, we use the *Joint Sentiment Topic Model (JST)* [5] as the baseline model. In addition to perplexity, we also compare the sentiment accuracy of SMSTM against JST on a test set of tweets with known sentiment polarity obtained from [1]. The sentiment accuracy indicates how well the sentiment prediction by a model aligns with the human judgement. A high sentiment accuracy in topic model is an indicator of how well a model can incorporate the sentiment in the generative process.

Perplexity Comparison

As defined in Chapter 4, a lower perplexity score of a model indicates a better predictive performance of a model. The perplexity of SMSTM for a test set can be calculated as:

$$\begin{aligned} &Perp(D^{SMSTM}) = \\ &\qquad \frac{1}{\sum_{u=1}^{U}\sum_{t=1}^{T} N_{ut}} \sum_{u=1}^{U}\sum_{t=1}^{T} log\Big(\sum_{k=1}^{K}\sum_{s=1}^{S} \theta_{u,k}\psi_{k,s}\Big(\sum_{n=1}^{N_w^{ut}} \pi_{s,k,1}\phi_{k,s,n} \\ &\qquad + \sum_{n=1}^{N_h^{ut}} \pi_{s,k,0}\eta_{k,s,n}\Big)\Big) \end{aligned} \tag{5.8}$$

We compare the perplexity of SMSTM against JST for different values of K ranging from 5 to 100. As it can seen from Figure 5.2, SMSTM clearly has a lower perplexity than JST, which indicates that SMSTM has a better predictive performance than JST on social media dataset.

[1] http://www.sentiment140.com/

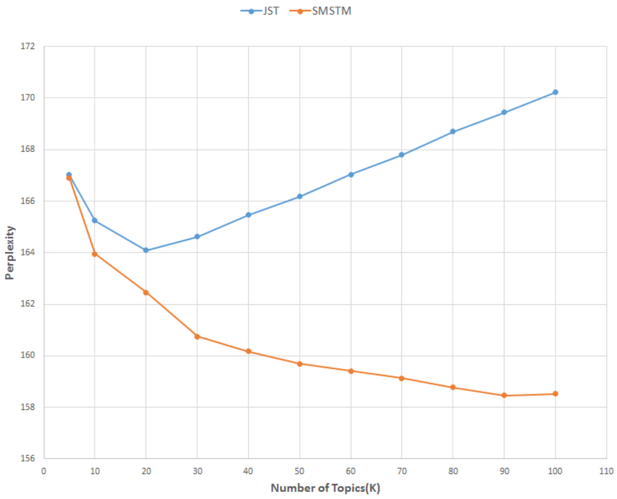

Figure 5.2: Perplexity comparison of SMSTM with JST

Sentiment Accuracy

To quantitatively evaluate the sentiment prediction attribute of SMSTM, we compare the sentiment accuracy of SMSTM against JST. In SMSTM, since sentiment s is a document-level parameter, we just use the value of s as the sentiment of the test tweet. For JST, the sentiment can be obtained by taking the maximum likelihood estimate of the variable π_d for each test tweet d. The comparison is shown in Figure 5.3.

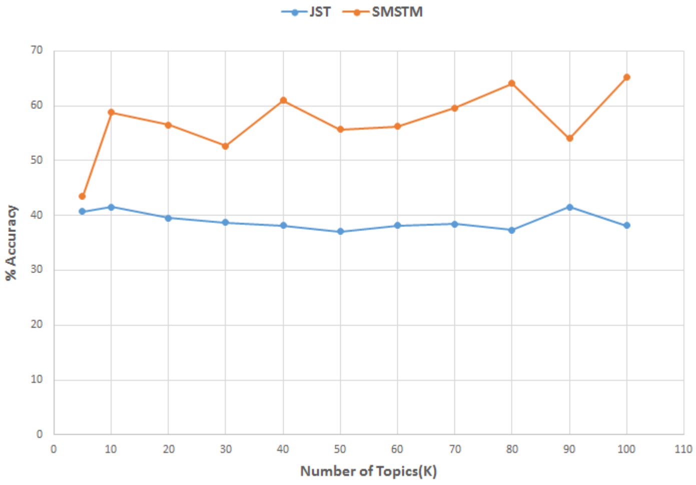

Figure 5.3: Sentiment accuracy comparison of SMSTM with JST

As it is evident from Figure 5.3, SMSTM clearly has a high sentiment accuracy than JST. JST has an accuracy of about 40% for nearly all values of K, whereas SMSTM shows a maximum sentiment accuracy of about 65%. This is because of the special treatment of the hyperparameter β in SMSTM, that makes it a better sentiment model than JST. This also shows that SMSTM can potentially be used as sentiment classifier tool.

5.5.4. Running time

Similar to SMTM, the running time per iteration of SMSTM also shows a similar trend, and increases with the number of topics K. This is a general trend observed in topic models.

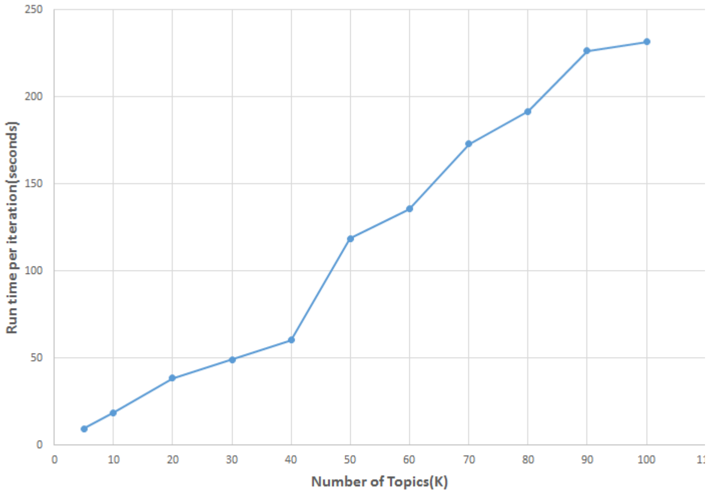

Figure 5.4: Running time per iteration for SMSTM

5.6. Conclusion

In this Chapter, we presented a sentiment topic model, namely SMSTM that can discover topics and their sentiments in social media data. We compared our model against the baseline JST model, and showed that SMSTM outperforms JST both qualitatively and quantitatively. Experimental results also suggest that SMSTM can potentially be used as a sentiment classifier for social media data.

6. Conclusion

6.1. Summary of Contributions

In this work, we presented two probabalistic models, namely SMTM and SMSTM to discover latent topics and sentiment in social media dataset. Both SMTM and SMSTM were based on the assumption that because of the short-length nature of social media text, all tokens in these posts belong to a single topic. Also, these models incorporate the special characteristics of these posts which is the hashtags. To the best of our knowledge, no previous work incorporates these 2 characteristic properties of social media datasets. SMSTM is able to determine the sentiment polarity of topics, and the associated sentiment-bearing polar words for different topics. One key outcome of SMSTM was that it was able to classify hashtags based on their sentiment polarity, without any training data for hashtag polarity. We evaluate both the models qualitatively and quantitatively, and found that both the models outperform the existing baseline techniques.

6.2. Scope

The models described in this work are designed for relatively short-length text that has both words and hashtags. The assumption of assigning a single topic and sentiment to all the words/tags in the document holds true only if the document is short in length. This is particularly true in case of social media data, where the length of the post is limited due to restrictions on the number of characters allowed, like *tweets*. If the documents are larger in length, this assumption might not hold true, and it might be better to assign topics at word or phrase level.

It is also suggested that topic models should be applied to preprocessed data. If the data is not preprocessed, stop words such as *for*, *the*, etc. might become dominant in the results, since the frequency of occurrence of these words is relatively high in any form of text as compared to other words.

6.3. General strengths and weaknesses of Bayesian models in topic modeling

The use of Bayesian models in topic modeling has both advantages and disadvantages.

Bayesian analysis allows taking into account the various uncertainties associated with the model parameters. It also provides a novel methodology to include the prior information associated with data in the model. This prior information can be combined with the new observations to give the posterior distribution of the data. It also provides a flexible and convenient way to model a wide variety of processes. Some of the models might have missing data, which can be modeled easily using Bayesian models. This is also accompanied by tractable inference techniques like Markov Chain Monte Carlo *MCMC* methods.

On the other hand, Bayesian models also have disadvantages. One of the main disadvantages is that it does not give a methodology to select the values of the prior information. It does not give a formal methodology to determine the prior knowledge into values of the hyperparameters. One key disadvantage of Bayesian models is the high computation cost. This cost is even higher when the number of latent parameters is large.

6.4. Directions for Future Work

A lot of work in the field of text analysis in social media data is exploratory in nature. A lot of vital information can be obtained from social media dataset which can be useful in many ways.

One approach to solve the problem of finding topics and sentiments in social media dataset could be to use Bayesin non-parametric techniques for modeling such data. Since SMSTM belongs to the family of parametric Bayesian models, it gives a set of both positive and negative words even for topics that are either only positive or negative. This problem could be solved using non-parametric models, by keeping the number of sentiments as variable for each topic.

In addition to text a lot of meta-data is embedded in social media data, that can be used in a variety of ways. One such attribute is the *location*, i.e. the geo-coordinates of the place from which this text originated. This can be used to determine the region-specific distribution of various topics, and to find region-specific attributes/words for each topic.

In terms of methodology, it might be interesting to relax the bag of words assumption in the model. Since sentiment also depends on the context in which the word is used, a better sentiment model could be developed using n-gram techniques. [16], [17] are some of the models that relax the bag-of-words assumption of topic models, but these models do not take into account the sentiment.

Finally, a more broader use of topic models can be made to solve the problem of community detection in social media datasets. [18] is one such work in this direction. A better community detection model can be developed that takes into account the links as well as the text posted by each user of the social network. The success of topic modeling techniques creates a lot of opportunities for their se in a variety of fields to tackle real world problems.

Bibliography

[1] D. M. Blei, A. Y. Ng and M. I. Jordan, *Latent Dirichlet Allocation*, Journal of Machine Learning Research 3 (2003) pp. 993–1022

[2] T. Hoffman, *Probabilistic latent semantic indexing*, In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, ACM pp. 50–57

[3] A. Pak and P. Paroubek, *Twitter as a Corpus for Sentiment Analysis and Opinion Mining*, LREC. Vol. 10. (2010) pp. 1320–1326

[4] A. Celikyilmaz, D. Hakkani-Tr and J, Feng, *Probabilistic model-based sentiment analysis of twitter messages* Spoken Language Technology Workshop (SLT), 2010 IEEE pp. 79–84

[5] C. Lin and Y. He, *Joint Sentiment/Topic model for Sentiment Analysis*, Proceedings of the 18th ACM conference on Information and Knowledge Management (2009) pp. 375–384

[6] Y. Jo and A. H. Oh, *Aspect and sentiment unification model for online review analysis*, Proceedings of the fourth ACM international conference on Web search and data mining, ACM (2011) pp. 815–824

[7] M. Rosen-Zvi, T. L. Griffiths, M. Steyvers and P. Smyth, *The Author-Topic Model for Authors and Documents*, Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence (2004) pp. 487–494

[8] L. Hong and B.D. Davidson, Empirical Study of Topic Modeling in Twitter, Proceedings of the First Workshop on Social Media Analytics (2010) pp. 80–88

[9] W. X. Zhao, J. Jiang, J, Weng, J, He, E. Lim, H. Yan and X. Li, *Comparing Twitter and Traditional Media Using Topic Models*, Advances in Information Retrieval. Springer Berlin Heidelberg (2011) pp. 338–349

[10] H. Zhang, B. Qiu, C.L. Giles, H.C. Foley and J. Yen, *An LDA-based community structure discovery approach for large-scale social networks*, In Intelligence and Security Informatics, 2007 IEEE pp. 200–207

[11] T. L. Griffiths and M. Steyvers, *Finding scientific topics*, Griffiths, Thomas L., and Mark Steyvers. "Finding scientific topics." Proceedings of the National Academy of Sciences 101, no. suppl 1 (2004) pp. 5228–523

[12] C. J. Hutto and E. Gilbert, *VADER: A parsimonious Rule-based Model for Sentiment Analysis of Social Media Text*, Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media (2014)

[13] A. Esuli anf F. Sebastian, *SentiWordNet: A Publicly Available Lexicon Resource for Opinion Mining*, Proceedings of 5th International Conference on Language Resouces and Evaluation, Genoa (2006) pp. 417–422

[14] T. Wilson, J, Wiebe and P. Hoffman, *Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis*, Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Laguage Processing (2005) pp. 347–354

[15] F. Li, M. Huang and X. Zhu, *Sentiment Analysis with Global Topics and Local Dependency*, AAAI. Vol. 10 (2010) pp. 1371–1376

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

[17] X. Wang, A. McCallum and X. Wei, *Topical n-grams: Phrase and topic discovery, with an application to information retrieval*, In ICDM 2007, Seventh IEEE International Conference on Data Mining pp. 697–702

[18] M. Sachan, A. Dubey, S. Shrivastava, E. P. Xing and E. Hovy, *Spatial compactness meets topical consistency: jointly modeling links and content for community detection*, In Proceedings of the 7th ACM international conference on Web search and data mining, ACM pp. 503–512

[19] Y. He, C. Lin, W. Gao and K.F. Wong *Dynamic joint sentiment-topic model*, ACM Transactions on Intelligent Systems and Technology (TIST) 5.1 (2013): 6

[20] F. Li, M. Huang and X. Zhu, *Sentiment Analysis with Global Topics and Local Dependency*, AAAI. Vol. 10 (2010) pp. 1371–1376

[21] W. Wei, K. Joseph and K. M. Carley, *A Bayesian graphical model to discover latent events from twitter*, Proceedings of the 9th The International AAAI Conference on Web and Social Media (2015)

[22] B. O'Connor, M. Krieger and D. Ahn, *TweetMotif: Exploratory Search and Topic Summarization for Twitter*, In Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media (2010)

[23] J. Eisenstein, B. O'Connor, N. A. Smith and E. P. Xing, *Diffusion of lexical change in social media*, (2014)

[24] F. Morstatter, L. Wu, T. H. Nazer, M. Karlsrud, K. M. Carley, H. Liu, *A New Approach to Bot Detection: The Importance of Recall*, In Advances in social networks analysis and mining (ASONAM), IEEE

Appendices

A. Derivation of Gibbs Sampling Equation for SMTM

The joint probability distribution of SMTM after integrating the parameters θ, π, ϕ and η is:

$$P(\boldsymbol{Z}, \boldsymbol{W}, \boldsymbol{C} | \alpha, \beta, \epsilon, \gamma) =$$
$$\prod_{u=1}^{U} \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \frac{\prod_{i=1}^{K} \Gamma(N_u^i + \alpha_i)}{\Gamma(\sum_{i=1}^{K} N_u^i + \alpha_i)} \prod_{i_1=1}^{K} \frac{\Gamma(\sum_{r=1}^{W} \beta_r)}{\prod_{r=1}^{W} \Gamma(\beta_r)} \frac{\prod_{r=1}^{W} \Gamma(M_{w_r}^{i_1} + \beta_r)}{\Gamma(\sum_{r=1}^{W} M_{w_r}^{i_1} + \beta_r)}$$
$$\prod_{i_2=1}^{K} \frac{\Gamma(\sum_{r=1}^{H} \epsilon_r)}{\prod_{r=1}^{H} \Gamma(\epsilon_r)} \frac{\prod_{r=1}^{H} \Gamma(M_{h_r}^{i_2} + \epsilon_r)}{\Gamma(\sum_{r=1}^{H} M_{h_r}^{i_2} + \epsilon_r)} \prod_{i_3=1}^{K} \frac{\Gamma(\sum_{r=0}^{1} \gamma_r)}{\prod_{r=0}^{1} \Gamma(\gamma_r)} \frac{\prod_{r=0}^{1} \Gamma(C_r^i + \gamma_r)}{\Gamma(\sum_{r=0}^{1} C_r^i + \gamma_r)}$$

To sample z_{ab}, we need $P(z_{ab} | \boldsymbol{Z_{-ab}}, \boldsymbol{C}, \boldsymbol{W}, \alpha, \beta, \gamma, \epsilon)$

$$P(z_{ab} = k | \boldsymbol{Z_{-ab}}, \boldsymbol{C}, \boldsymbol{W}, \alpha, \beta, \gamma, \epsilon) \propto P(z_{ab} = k, \boldsymbol{Z_{-ab}}, \boldsymbol{C}, \boldsymbol{W}, \alpha, \beta, \gamma, \epsilon)$$

$$= \left(\frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)}\right)^{U} \left(\prod_{u \neq a} \frac{\prod_{i=1}^{K} \Gamma(N_u^i + \alpha_i)}{\Gamma(\sum_{i=1}^{K} N_u^i + \alpha_i)}\right) \frac{\prod_{i=1}^{K} \Gamma(N_a^i + \alpha_i)}{\Gamma(\sum_{i=1}^{K} N_a^i + \alpha_i)}$$
$$\left(\frac{\Gamma(\sum_{r=1}^{W} \beta_r)}{\prod_{r=1}^{W} \Gamma(\beta_r)}\right)^{K} \left(\prod_{i_i=1}^{K} \prod_{r \notin W_{ab}} \Gamma(M_{w_r}^{i_1} + \beta_r)\right) \left(\prod_{i_i=1}^{K} \frac{\prod_{r \in W_{ab}} \Gamma(M_{w_r}^{i_1} + \beta_r)}{\Gamma(\sum_{r=1}^{W} M_{w_r}^{i_1} + \beta_r)}\right)$$
$$\left(\frac{\Gamma(\sum_{r=1}^{H} \epsilon_r)}{\prod_{r=1}^{H} \Gamma(\epsilon_r)}\right)^{K} \left(\prod_{i_2=1}^{K} \prod_{r \notin H_{ab}} \Gamma(M_{h_r}^{i_2} + \epsilon_r)\right) \left(\prod_{i_2=1}^{K} \frac{\prod_{r \in H_{ab}} \Gamma(M_{h_r}^{i_2} + \epsilon_r)}{\Gamma(\sum_{r=1}^{H} M_{h_r}^{i_2} + \epsilon_r)}\right)$$
$$\left(\frac{\Gamma(\sum_{r=0}^{1} \gamma_r)}{\prod_{r=0}^{1} \Gamma(\gamma_r)}\right)^{K} \left(\prod_{i_3=1}^{K} \frac{\prod_{r=0}^{1} \Gamma(C_r^{i_3} + \gamma_r)}{\Gamma(\sum_{r=0}^{1} C_r^{i_3} + \gamma_r)}\right)$$

$$\propto \frac{\prod_{i=1}^{K} \Gamma(N_a^i + \alpha_i)}{\Gamma(\sum_{i=1}^{K} N_a^i + \alpha_i)} \quad \left(\prod_{i_i=1}^{K} \frac{\prod_{r \in W_{ab}} \Gamma(M_{w_r}^{i_1} + \beta_r)}{\Gamma(\sum_{r=1}^{W} M_{w_r}^{i_1} + \beta_r)}\right)$$
$$\left(\prod_{i_2=1}^{K} \frac{\prod_{r \in H_{ab}} \Gamma(M_{h_r}^{i_2} + \epsilon_r)}{\Gamma(\sum_{r=1}^{H} M_{h_r}^{i_2} + \epsilon_r)}\right) \left(\prod_{i_3=1}^{K} \frac{\prod_{r=0}^{1} \Gamma(C_r^{i_3} + \gamma_r)}{\Gamma(\sum_{r=0}^{1} C_r^{i_3} + \gamma_r)}\right)$$

For every variable x, let the notation x^{-ab} denote the same number as x, but with $z_{a,b}$ excluded. Then for quantities that depend on $z_{a,b}$,

$$x = x^{-ab} + 1$$

Then the above equation can be written as:

$$= \left(\prod_{i \neq k} \Gamma(N_a^{i,-ab} + \alpha_i)\right) \frac{\Gamma(N_a^{k,-ab} + \alpha_k + 1)}{\Gamma((\sum_{i=1}^{K} N_a^{i,-ab} + \alpha_i) + 1)}$$
$$\left(\prod_{i_i \neq K} \frac{\prod_{r \in W_{ab}} \Gamma(M_{w_r}^{i_1,-ab} + \beta_r)}{\Gamma(\sum_{r=1}^{W} M_{w_r}^{i_1,-ab} + \beta_r)}\right) \left(\frac{\prod_{r \in W_{ab}} \Gamma(M_{w_r}^{k,-ab} + \beta_r + n_{ab}^{w,r})}{\Gamma(\sum_{r=1}^{W} M_{w_r}^{k,-ab} + \beta_r + n_{ab}^{w,(.)})}\right)$$
$$\left(\prod_{i_2 \neq K} \frac{\prod_{r \in H_{ab}} \Gamma(M_{h_r}^{i_2,-ab} + \epsilon_r)}{\Gamma(\sum_{r=1}^{H} M_{h_r}^{i_2,-ab} + \epsilon_r)}\right) \left(\frac{\prod_{r \in H_{ab}} \Gamma(M_{h_r}^{k,-ab} + \epsilon_r + n_{ab}^{h,r})}{\Gamma(\sum_{r=1}^{H} M_{h_r}^{k,-ab} + \epsilon_r + n_{ab}^{h,(.)})}\right)$$
$$\left(\prod_{i_3 \neq K} \frac{\prod_{r=0}^{1} \Gamma(C_r^{i_3,-ab} + \gamma_r)}{\Gamma(\sum_{r=0}^{1} C_r^{i_3,-ab} + \gamma_r)}\right) \left(\frac{\prod_{r=0}^{1} \Gamma(C_r^{k,-ab} + \gamma_r + n_{ab}^{r,(.)})}{\Gamma((\sum_{r=0}^{1} C_r^{k,-ab} + \gamma_r) + n_{ab}^{(.),(.)})}\right)$$

By using the property of Gamma function

$$\Gamma(x + 1) = x\Gamma(x)$$

we can split and then combine to simplify the equation as:

$$= \frac{\prod_{i=1}^{K} \Gamma(N_a^{i,-ab} + \alpha_i)}{\Gamma(\sum_{i=1}^{K} N_a^{i,-ab} + \alpha_i)} \frac{N_a^{k,-ab} + \alpha_k}{\sum_{i=1}^{K} N_a^{i,-ab} + \alpha_i}$$

$$\left(\prod_{i_i=1}^{K} \frac{\prod_{r\in W_{ab}} \Gamma(M_{w_r}^{i_1,-ab}+\beta_r)}{\Gamma(\sum_{r=1}^{W} M_{w_r}^{i_1,-ab}+\beta_r)}\right) \left(\frac{\prod_{r\in W_{ab}} \prod_{j=0}^{n_{ab}^{w,r-1}} (M_{w_r}^{k,-ab}+\beta_r+j)}{\prod_{j=0}^{n_{a,b}^{w,(.)-1}} ((\sum_{r=1}^{W} M_{w_r}^{k,-ab}+\beta_r)+j)}\right)$$

$$\left(\prod_{i_2=1}^{K} \frac{\prod_{r\in H_{ab}} \Gamma(M_{h_r}^{i_2,-ab}+\epsilon_r)}{\Gamma(\sum_{r=1}^{H} M_{h_r}^{i_2,-ab}+\epsilon_r)}\right) \left(\frac{\prod_{r\in H_{ab}} \prod_{j=0}^{n_{ab}^{h,r-1}} (M_{h_r}^{k,-ab}+\epsilon_r+j)}{\prod_{j=0}^{n_{a,b}^{h,(.)-1}} ((\sum_{r=1}^{H} M_{h_r}^{k,-ab}+\epsilon_r)+j)}\right)$$

$$\left(\prod_{i_3=1}^{K} \frac{\prod_{r=0}^{1} \Gamma(C_r^{i_3,-ab}+\gamma_r)}{\Gamma(\sum_{r=0}^{1} C_r^{i_3,-ab}+\gamma_r)}\right) \left(\frac{\prod_{r=0}^{1} \prod_{j=0}^{n_{ab}^{r,(.)}-1} (C_r^{k,-ab}+\gamma_r+j)}{\prod_{j=0}^{n_{ab}^{(.),(.)}-1} ((\sum_{r=0}^{1} C_r^{k,-ab}+\gamma_r)+j)}\right)$$

$$\propto \frac{N_a^{k,-ab}+\alpha_k}{\sum_{i=1}^{K} N_a^{i,-ab}+\alpha_i} \left(\frac{\prod_{r\in W_{ab}} \prod_{j=0}^{n_{ab}^{w,r-1}} (M_{w_r}^{k,-ab}+\beta_r+j)}{\prod_{j=0}^{n_{a,b}^{w,(.)-1}} ((\sum_{r=1}^{W} M_{w_r}^{k,-ab}+\beta_r)+j)}\right)$$

$$\left(\frac{\prod_{r\in H_{ab}} \prod_{j=0}^{n_{ab}^{h,r-1}} (M_{h_r}^{k,-ab}+\epsilon_r+j)}{\prod_{j=0}^{n_{a,b}^{h,(.)-1}} ((\sum_{r=1}^{H} M_{h_r}^{k,-ab}+\epsilon_r)+j)}\right) \left(\frac{\prod_{r=0}^{1} \prod_{j=0}^{n_{ab}^{r,(.)}-1} (C_r^{k,-ab}+\gamma_r+j)}{\prod_{j=0}^{n_{ab}^{(.),(.)}-1} ((\sum_{r=0}^{1} C_r^{k,-ab}+\gamma_r)+j)}\right)$$

B. Derivation of Gibbs Sampling Equation for SMSTM

The joint probability distribution for SMSTM after integrating the parameters θ, π, ϕ, η and ψ can be given as:

$$
\begin{aligned}
&P(\boldsymbol{Z}, \boldsymbol{S}, \boldsymbol{W}, \boldsymbol{C} | \alpha, \beta, \epsilon, \gamma, \lambda) = \\
&\prod_{u=1}^{U} \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \frac{\prod_{i=1}^{K} \Gamma(N_u^i + \alpha_i)}{\Gamma(\sum_{i=1}^{K} N_u^i + \alpha_i)} \\
&\prod_{i_1=1}^{K} \prod_{s_1=1}^{S} \frac{\Gamma(\sum_{r=1}^{W} \beta_{r_s})}{\prod_{r=1}^{W} \Gamma(\beta_{r_s})} \frac{\prod_{r=1}^{W} \Gamma(M_{w_r}^{i_1,s_1} + \beta_r)}{\Gamma(\sum_{r=1}^{W} M_{w_r}^{i_1,s_1} + \beta_r)} \\
&\prod_{i_2=1}^{K} \prod_{s_2=1}^{S} \frac{\Gamma(\sum_{r=1}^{H} \epsilon_r)}{\prod_{r=1}^{H} \Gamma(\epsilon_r)} \frac{\prod_{r=1}^{H} \Gamma(M_{h_r}^{i_2,s_2} + \epsilon_r)}{\Gamma(\sum_{r=1}^{H} M_{h_r}^{i_2,s_2} + \epsilon_r)} \\
&\prod_{i_3=1}^{K} \prod_{s_3=1}^{S} \frac{\Gamma(\sum_{r=0}^{1} \gamma_r)}{\prod_{r=0}^{1} \Gamma(\gamma_r)} \frac{\prod_{r=0}^{1} \Gamma(C_r^{i_3,s_3} + \gamma_r)}{\Gamma(\sum_{r=0}^{1} C_r^{i_3,s_3} + \gamma_r)} \\
&\prod_{i_4=1}^{K} \frac{\Gamma(\sum_{s=0}^{S} \lambda_s)}{\prod_{s=0}^{S} \Gamma(\lambda_r)} \frac{\prod_{s=0}^{S} \Gamma(L^{i_4,s} + \lambda_s)}{\Gamma(\sum_{s=0}^{S} L^{i_4,s} + \lambda_s)}
\end{aligned}
$$

To sample z_{ab} and s_{ab}, we need $P(z_{ab}, s_{ab} | \boldsymbol{Z_{-ab}}, \boldsymbol{S_{-ab}}, \boldsymbol{C}, \boldsymbol{W}, \alpha, \beta, \epsilon, \gamma, \lambda)$

$$
\begin{aligned}
&P(z_{ab} = k, s_{ab} = p | \boldsymbol{Z_{-ab}}, \boldsymbol{S_{-ab}}, \boldsymbol{C}, \boldsymbol{W}, \alpha, \beta, \epsilon, \gamma, \lambda) \\
&\propto P(z_{ab} = k, s_{ab} = p, \boldsymbol{Z_{-ab}}, \boldsymbol{S_{-ab}}, \boldsymbol{C}, \boldsymbol{W}, \alpha, \beta, \epsilon, \gamma, \lambda)
\end{aligned}
$$

The joint distribution here is similar to SMTM, except that we have an additional term that was generated by integrating the parameter ψ. The Gibbs sampling here will be similar to the sampling in SMTM, with one additional term.

The sampling formula will be:

$$
\begin{aligned}
&P(z_{ab} = k, s_{ab} = p, \boldsymbol{Z_{-ab}}, \boldsymbol{S_{-ab}}, \boldsymbol{C}, \boldsymbol{W}, \alpha, \beta, \epsilon, \gamma, \lambda) \propto \\
&\frac{N_{u,(.)}^{k,-ut} + \alpha_k}{\sum_{i=1}^{K} N_{u,(.)}^{i,-ut} + \alpha_i} \frac{L^{k,p,-ut} + \lambda_p}{\sum_{s=0}^{1} L^{k,s,-ut} + \lambda_s} \frac{\prod_{r \in W_{ut}} \prod_{j=0}^{n_{ut}^{w,r}-1} (M_{w_r}^{k,p,-ut} + \beta_r + j)}{\prod_{j=0}^{n_{ut}^{w,(.)}-1} ((\sum_{r=1}^{W} M_{w_r}^{k,p,-ut} + \beta_r) + j)} \\
&\frac{\prod_{r \in H_{ut}} \prod_{j=0}^{n_{ut}^{h,r}-1} (M_{h_r}^{k,p,-ut} + \epsilon_r + j)}{\prod_{j=0}^{n_{ut}^{h,(.)}-1} ((\sum_{r=1}^{H} M_{h_r}^{k,p,-ut} + \epsilon_r) + j)} \frac{\prod_{r=0}^{1} \prod_{j=0}^{n_{ut}^{r,(.)}-1} (C_r^{k,p,-ut} + \gamma_r + j)}{\prod_{j=0}^{n_{ut}^{(.),(.)}-1} ((\sum_{r=0}^{1} C_r^{k,p,-ut} + \gamma_r) + j)}
\end{aligned}
$$

Institute for Software Research • Carnegie Mellon University • 5000 Forbes Avenue • Pittsburgh, PA 15213-3890