Large Scale Microblog Mining Using Distributed MB-LDA

Chenyi Zhang
College of Computer Science
Zhejiang University
Hangzhou, Zhejiang, 310027
P.R.China
zhangchenyi.zju@gmail.com

Jianling Sun
College of Computer Science
Zhejiang University
Hangzhou, Zhejiang, 310027
P.R.China
sunjl@zju.edu.cn

ABSTRACT

In the information explosion era, large scale data processing and mining is a hot issue. As microblog grows more popular, microblog services have become information provider on a web scale, so researches on microblog begin to focus more on its content mining than solely user's relationship analysis before. Although traditional text mining methods have been studied well, no algorithm is designed specially for microblog data, which contain structured information on social network besides plain text. In this paper, we introduce a novel probabilistic generative model MicroBlog-Latent Dirichlet Allocation (MB-LDA), which takes both contactor relevance relation and document relevance relation into consideration to improve topic mining in microblogs. Through Gibbs sampling for approximate inference of our model, MB-LDA can discover not only the topics of microblogs, but also the topics focused by contactors. When faced with large datasets, traditional techniques on single node become less practical within limited resources. So we present distributed MB-LDA in MapReduce framework in order to process large scale microblogs with high scalability. Furthermore, we apply a performance model to optimize the execution time by tuning the number of mappers and reducers. Experimental results on actual dataset show MB-LDA outperforms the baseline of LDA and distributed MB-LDA offers an effective solution to topic mining for large scale microblogs.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications;

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence

General Terms

Algorithms, Performance, Experimentation

Keywords

Distributed MB-LDA, Large-scale, Microblogs, Social network, MapReduce

1. INTRODUCTION

Microblog has become a popular social networking service in the recent years because of its ease of use and convenience. Based on user's social relationship, microblog services offer a real-time platform to update personal status and share interesting things with friends on their buddy lists. Microbloggers can publish their microblogs over multiple delivery channels, e.g., via web interface, cell phone or other user applications. Famous microblog service Twitter, as an information provider on a web scale, has reached a new milestone: Twitter users are now sending over 200 million tweets per day, which is huge in data volume.

Microblogs can be divided into three categories[1]: Broadcast, conversation, or retweet messages. Broadcast messages are most common and can be seen by any users; conversation messages, starting with a special symbol "@", have specific contactors to talk to; retweet messages, identified by a special symbol "RT", allow users to forward other's messages with their personal comments.

In the information explosion era, how to effectively dig out latent topics and internal semantic structures from large scale data is an important issue. Microblogs contain structured information on social network besides plain text, and the relationships on social network can play a supporting role in topic mining. On the other hand, microblogs are short text carried with limited information (restricted to 140 characters), which will increase the difficulty of topic mining. These natural features of microblogs mentioned above prevent traditional text mining algorithms to be employed directly with their full potentials.

Early work of microblog mainly focused on its user relationship and community structure. For example, Java et al. [2] studied the topological and geographical properties of Twitter's social network. Others such as Krishnamurthy et al.[3] studied user behaviors and geographic growth patterns of Twitter. Only a few researches on content analysis of microblog were proposed recently[4], that is mainly because traditional text mining algorithms, which are suitable for old corpora, cannot model microblog data very well without considering its inner structured information on social network. In this paper, we proposed a novel probabilistic generative model based on LDA, called MB-LDA, which takes both contactor relevance relation and document relevance relation into consideration to improve topic mining in microblogs. In real world applications, as large amount of microblogs are increasing day by day, traditional techniques on single node become less practical within limited resources. So we present distributed MB-LDA in MapReduce framework[5] in order to process large scale microblogs with high scalability. Further optimization is carried out to speed up the execution time by tuning the number of mappers and reducers.

WWW 2012 Companion, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.

In this paper, we make the following contributions:

- A novel model MB-LDA is proposed, which is suitable for microblog data by taking both structured information (i.e. contactor relation, forwarding relation) and unstructured information (i.e. text) into consideration.
- Distributed MB-LDA in MapReduce framework is proposed in order to meet the requirement of processing large scale microblogs with high scalability.
- Experimental results on actual dataset show MB-LDA outperforms the baseline of LDA and distributed MB-LDA offers an effective solution to topic mining for large scale microblogs.

The rest of the paper is organized as follows. Section 2 introduces the previous work related to this paper. Section 3 introduces the novel model MB-LDA for microblog mining. Section 4 proposes the distributed MB-LDA in MapReduce framework for large scale situations and applies performance model to further optimize our approach. The experiment is conducted in Section 5, which is followed by the conclusion of our contribution in Section 6.

2. RELATED WORK

There exist many algorithms for topic mining for text in literature. The clustering methods are early solution to this problem, which transfer unstructured text data to vectors by Vector Space Model (VSM) and do clustering with traditional methods like K-means[6]. Clustering results are usually considered sharing the same topic respectively. However, the major disadvantage of clustering methods is that many of these algorithms depend on distance functions for the pairwise distance measurements, which are difficult to define in large scale corpora; besides, there lies no semantic information in clustering results, which need further analysis to extract topics.

Dimensionality reduction method like Latent Semantic Analysis (LSA) was introduced to text mining by Deerwester et al.[7] By assuming that words close in meaning will occur close together in text, LSA constructs a term-document matrix in the popular tf-idf scheme[8] and use singular value decomposition to capture its latent semantic in the concept space[9]. Although LSA can extract topics from corpora and find relations between terms in a semantic way, the limitation is that the result of SVD is less interpretable and LSA itself cannot capture polysemy. LSA is not suitable to large scale text mining due to its high computing cost.

Probabilistic topic models such as Latent Dirichlet Allocation (LDA) was introduced to text modeling by Blei et al.[10] LDA is also a method to recover the latent topic structure, which extends PLSA[11] by defining a complete probabilistic generative model. The intuition behind LDA is that documents exhibit multiple topics which are represented by distributions over words. In the framework of LDA, words of documents are the observed variables while topic structures are the hidden variables. Through probabilistic inference for LDA, the hidden variables are inferred and topics can be discovered from the corpus[12].

Topic model is widely applied to topic mining[10], information retrieval[13], text classification[14] and social network analysis[15]. Ideas associated to topic model are also adapted to non-text domains[16] such as image and music. There are also some extension models considering relationship between documents, for example, Link-PLSA-LDA and Hypertext Topic Model (HTM), which are relevant to our work: Link-PLSA-LDA was introduced for citation analysis by Nallapati et al[17], assuming that cited document is generated by PLSA, citing document is generated by LDA and two documents share the same topic; HTM was proposed to process hypertexts by Congkai Sun et al[18], which takes hyperlinks into consideration and outperforms the baseline methods on topic mining and text classification.

In recent years, with the rapid increase in the amount of published information and the effects of this abundance of data, parallel machine learning methods were developed to satisfy the requirement of large scale text processing, especially after the MapReduce[5] technology was introduced by Google to support distributed computing on large data sets on clusters of computers with high scalability. Relevant parallel methods for LDA include the following: Newman et al.[19] proposed Approximate Distributed LDA (AD-LDA), where topic-document matrix is stored locally, each processor performs a local Gibbs sampling step, followed by a step of synchronized global update for topic-word matrix. Asuncion et al.[20] proposed asynchronous distributed LDA, where the update step changes to an asynchronous communication with another random processor. Wang et al.[21] proposed an MPI implementation of AD-LDA, which can be applied to real world applications such as communication recommendation systems. Smola et al.[22] proposed a high performance sampling architecture for inference of latent topic models on a cluster of workstations.

α_d α_c	Hyperparameters for θ_d and θ_c
β	Hyperparameters for φ
T	Number of topics
D	Number of microblogs
V	Number of words
c	Contactors in conversation messages(@)
λ	Weight hyperparameters for retweet messages
θ_c	Topic distribution associated with contactor c
θ_d	Topic distribution over microblog d
$\theta_{d_{RT}}$	Topic distribution over retweet message d
φ	Word distribution over topics
z_i z_{-i}	Topic of word i (indicators before sampling i)
w	Words in microblogs
$n_{d,j}$	Number of times topic j is assigned to d
$n_{d,\cdot}$	Number of times all topics are assigned to d
$n_{c,j}$	Number of times topic j is focused by c
$n_{c,\cdot}$	Number of times all topics are focused by c
$n_{j,v}$	Number of times word v is assigned to topic j
$n_{j,\cdot}$	Number of times all words are assigned to topic j
π_c	Decision whether is a conversation message
r	Decision whether is a retweet message

Fig. 1 Notation

3. MicroBlog Latent Dirichlet Allocation

3.1 Modeling

Different from plain text, microblog has its special symbols (i.e. @ and RT) to characterize the relation between microblogs: @ indicates the contactor relevance relation of microblogs and RT indicates the document relevance relation of microblogs, which are defined as follows.

Definition 1. ***Contactor relevance relation*** *of microblogs refers to that conversation message and its contactor (@) have latent semantic relationships. In general, messages with the same contactor usually share the related topics.* This phenomenon is very common in conversation microblogs. For example, two conversation messages: "@Ethan Can you lend me a book on data mining" and "@Ethan HELP me on these computer exercises", if considering the contactor relevance relation, we can set a connection on these two seemingly unrelated microblogs and infer that "computer exercises" in the latter message are related to data mining.

Definition 2. ***Document relevance relation*** *of microblogs refers to that comment on retweet message and its original content have latent semantic relationships. In general, the comment part and original part share the related topics.* This phenomenon is very common in retweet microblogs. For example, one retweet message: "Good job RT @Ethan I have finished this experiment", we can hardly dig out topics from the simple comment "Good job". But by considering the document relevance relation, we can infer that "Good job" is an experiment work.

MB-LDA is a unified model for topic mining of microblogs based on LDA, by overall considering contactor relevance relation and document relevance relation. MB-LDA adopts the basic idea of topic model, namely each microblog exhibits multiple topics which are represented by probability distributions over words, denoted as $P(z|w)$ and $P(z|w)$ respectively. The Bayesian network of MB-LDA is showed in Fig. 2.

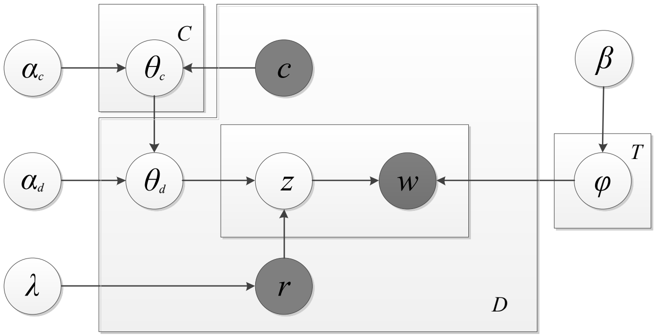


Fig. 2 Bayesian network of MB-LDA

MB-LDA generates microblogs in the following process:

1. Random choose a topic distribution over words. The distribution φ is sampled from a Dirichlet distribution with hyperparameter β.
2. Judge whether is a conversation microblog according to the symbol "@". If so, mark π_c as 1, random choose a contactor-topic distribution θ_c, which is sampled from a Dirichlet distribution with hyperparameter α_c, then assign the value of θ_c to θ_d; if not, random choose a document-topic distribution θ_d, which is sampled from a Dirichlet distribution with hyperparameter α.

The probability distribution of θ is showed as follows:

$$P(\theta|\alpha)=P(\theta|\alpha,c)=P(\theta_c|\alpha_c)^{\pi_c}P(\theta_d|\alpha_d)^{1-\pi_c} \qquad (1)$$

3. Judge whether is a retweet microblog according to the symbol "RT". If so, the distribution of forwarding part d_{RT} over topics is $\theta_{d_{RT}}$, sample r from a Bernoulli distribution with parameter λ to decide from which Multinomial distribution (with parameter $\theta_{d_{RT}}$or θ_d) to draw the topic assignment z_{dn} of current word; if not, set r as 0, straightforward draw the topic assignment z_{dn} of current word from the Multinomial distribution with parameter θ_d.
4. Draw the specific word w_{dn} from the Multinomial distribution with parameter $\varphi_{z_{dn}}$.

For a microblog, the joint probability distribution over all words and their topics is showed as follows:

$$\begin{aligned}P(\boldsymbol{w},\boldsymbol{z}|\lambda,\theta,\beta)&=P(r|\lambda)P(\boldsymbol{z}|\theta)P(\boldsymbol{w}|\boldsymbol{z},\beta)\\&=P(r|\lambda)P(\boldsymbol{z}|\theta_d)^{1-r}P(\boldsymbol{z}|\theta_{d_{RT}})^{r}P(\boldsymbol{w}|\boldsymbol{z},\beta)\end{aligned} \qquad (2)$$

Formal description of generative process in MB-LDA is as follows:

```
for each topic $k \in \{1,2...T\}$ do
    draw $\varphi_k \sim Dir(\beta)$
end for
for each microblog $d$ do
  use "@" to choose a contact
  if starts with "@"
    draw $\theta_d = \theta_c \sim Dir(\alpha_c)$
  else
    draw $\theta_d \sim Dir(\alpha_d)$
  for each word $w_{dn}$ do
    use "RT" to judge a relation
    if has "RT"
      draw $r = Ber(\lambda)$
      if $r$=1
        draw $z_{dn} \sim Multi(\theta_{d_{RT}})$
      else
        $z_{dn} \sim Multi(\theta_d)$
    else
      draw $z_{dn} \sim Multi(\theta_d)$
    draw $w_{dn} \sim Multi(\varphi_{z_{dn}})$
  end for
end for
```

Fig. 3 Generative process of microblogs

3.2 Inference

As showed in Fig. 2, the words in microblogs, contactor relation and retweet relation are observed while the topic structure (i.e. the topics, document-topic distribution, etc.) is hidden structure. The central problem for topic models is to infer the hidden variables using the observed ones (computing the posterior distribution of the hidden variables given the observed variables), which can be thought of as "reversing" the generative process.

Inference methods for topic models include variational Bayesian[10], Gibbs sampling[23], expectation propagation[24]. Among them, Gibbs sampling, a special case of Markov chain Monte Carlo, is a fast and effective algorithms for approximate inference in high-dimensional models. In this paper, we adopt Gibbs sampling to infer the hidden structure of MB-LDA.

The inference process of MB-LDA is as follows:

1. Expand Equation (1) and (2) by Euler integration:

$$P(\boldsymbol{w} \mid \boldsymbol{z}, \beta) = \left(\frac{\Gamma(V\beta)}{\Pi_v \Gamma(\beta)} \right)^T \prod_{j=1}^{T} \frac{\Pi_v \Gamma(n_{j,v} + \beta)}{\Gamma(n_{j,\bullet} + V\beta)} \tag{3}$$

$$P(\boldsymbol{z} \mid \alpha) = \left(\frac{\Gamma(T\alpha)}{\Pi_j \Gamma(\alpha)} \right)^T \prod_{d=1}^{D} \frac{\Pi_j \Gamma(n_{d,j} + \alpha)}{\Gamma(n_{d,\bullet} + T\alpha)} \tag{4}$$

2. Sample posterior distribution using Gibbs sampling:

$$\begin{aligned} P(z_i = j \mid \boldsymbol{w}, \boldsymbol{z}_{-i}, \alpha, \beta) &= \frac{P(\boldsymbol{z}, \boldsymbol{w} \mid \alpha, \beta)}{P(\boldsymbol{z}_{-i}, \boldsymbol{w} \mid \alpha, \beta)} \\ &\propto \frac{n_{j,v} + \beta - 1}{n_{j,\bullet} + V\beta - 1} \times \frac{n_{d,j} + \alpha - 1}{n_{d,\bullet} + T\alpha - 1} \\ &\propto \frac{n_{j,v} + \beta - 1}{n_{j,\bullet} + V\beta - 1} \times (n_{d,j} + \alpha - 1) \end{aligned} \tag{5}$$

3. Iterate Equation (5) for all topics until it reaches the convergence for the entire microblog set.

The final results of θ_d and φ_z are listed as follows:

$$\theta_d = \frac{n_{d,j} + \alpha - 1}{n_{d,\bullet} + T\alpha - 1} \tag{6}$$

$$\varphi_z = \frac{n_{j,v} + \beta - 1}{n_{j,\bullet} + V\beta - 1} \tag{7}$$

Similarly, the final result of contactor-topics distribution θ_c is as follows:

$$\theta_c = \frac{n_{c,j} + \alpha_c - 1}{n_{c,\bullet} + T\alpha_c - 1} \tag{8}$$

Thus, according to these topic related distributions, MB-LDA can conclude the several high probability topics as hot topics in microblogs and find out the most representative words for each topic. Besides content mining, with the contactor-topics distribution, MB-LDA can dig out interesting topics focused by each contactor.

In summary, MB-LDA can discover not only the topics of microblogs, but also the topics focused by contactors. With the result of topic mining, we can conduct several personalized analysis and applications on microblogs, such as similar microblog detection, microblog recommendation, and social circle recommendation.

3.3 Extension

The idea of the model can be not only applied to topic mining for microblogs, but also extended to many text data with social links, e.g., Email. The reply relationship in Email is an analogy to the forwarding relationship (RT) in microblog while the To list and Cc list in Email are similar to the contactor (@) in microblog. Other application scenarios such as chat history mining, forum posts mining can also learn from MB-LDA model.

4. DISTRIBUTED MB-LDA

Distributed MB-LDA is proposed to handle the real world application with large scale microblogs when MB-LDA on single node becomes less practical within limited computing and storage resources. Distributed MB-LDA is developed in the framework of MapReduce, which is a software framework introduced by Google to support distributed computing on large data sets on clusters of computers.

4.1 MapReduce overview

MapReduce is a framework for processing huge datasets with distributed Map and Reduction operations (Fig. 4). In Map step, the master node splits input data into partitions, which can be processed by user-defined Map functions, and produces (key, value) tuples as intermediate output. In Reduce step, the Reduce functions merge all intermediate tuples with the same key, sort them and output the final (key, value) tuples. Both the map function and reduce function can be executed in parallel on non-overlapping input and intermediate data.

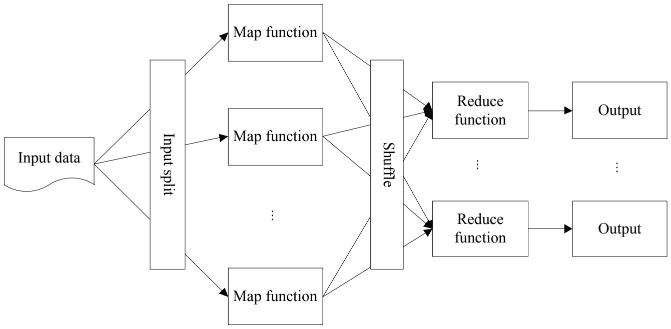


Fig. 4 MapReduce framework

4.2 Solution

In MapReduce framework, microblogs must be split into partitions. Under the situation of real world application, the number of microblogs is huge (over millions) as well as the number of contactors, while the totally number of words is relatively small (about 20 thousand). Equation (5) implies that only a global variable $n_{j,\cdot}$ is used in sampling. Through this analysis on the inference process of MB-LDA, we find out if microblogs data are split by contactor, document-topic count matrix and contactor-topic count matrix can be stored locally while word-topic count matrix must be stored globally because only word-topic count matrix needs global updates. As mentioned above, the word-topic count matrix is not space consuming so it can be stored in memory; and for document-topic count matrix and contactor-topic count matrix, they are so huge to restrict a single node computation, so they are stored in HDFS and can be accessed by row key. In each Gibbs sampling iteration, each processor unit p samples its local topic assignment z_p from the posterior distribution of Equation (5) in parallel, and then updates local stored matrices as well as global stored matrix according to the new topic assignments.

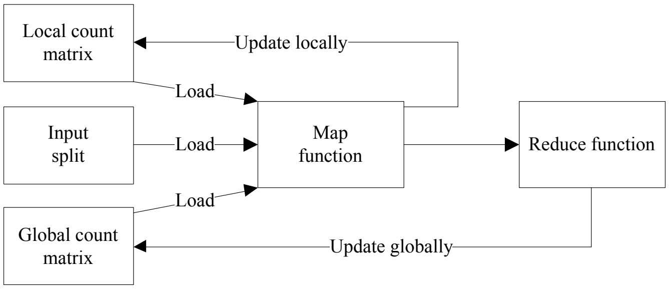


Fig. 5 Sampling iteration in distributed MB-LDA

We model each Gibbs sampling iteration of distributed MB-LDA as a MapReduce job, as illustrated in Figure 5, where the Map step conducts Gibbs sampling in parallel and the Reduce step updates the global count matrix incrementally for next iteration. The working process of distributed MB-LDA is depicted in Figure 6.

```
Initialization
    Split microblog data into partitions
    Randomly initialize the count matrices

Map step
    for each processor unit $p$ in parallel do
        Load global word-topic count matrix to local
        Parse key to the specific microblog
        Parse value to all words in the specific microblog
        Load local document-topic count matrix
        Load local contactor-topic count matrix
        Sample $z_p$ locally via MB-LDA inference
        Update local count matrices
    end for

Reduce step
    Update global word-topic count matrix
```

Fig. 6 Process of distributed MB-LDA

After enough iterations, the model reaches the convergence and calculates out the final count matrices. With these count matrices, we can obtain the related distributions by Equation (6-8), which are useful in topic mining for microblogs.

In MapReduce environment with a cluster's resource, the mappers mainly conduct the sampling process in parallel and the reducers mainly update the count matrices for next iteration, which makes distributed MB-LDA effective and scalable in topic mining for large scale microblogs.

4.3 Optimization

In order to accelerate the running time of distributed MB-LDA on MapReduce, we apply a performance model to further optimize our approach. This performance model[25] is designed for time optimization in MapReduce framework through a task pipeline by tuning the number of mappers and reducers.

The model concludes under the situation that the system capacity is M mappers and R reducers; each mapper has a constant overhead C_1; each reducer has a constant overhead C_2; network transfer rate is V_n; the size of intermediate data is S', the best scheduling plan is to set X mappers and Y reducers where $X = M\sqrt{\frac{C_2}{C_1} + \frac{S'}{C_1 R V_n}}$ and $Y = R$. According to this scheduling plan, we can further accelerate the overall running time of distributed MB-LDA.

5. EXPERIMENTS

5.1 Experiment setup

5.1.1 Dataset

In this paper, we use a microblog dataset, original from Twitter, to examine the performance of MB-LDA and distributed MB-LDA. This dataset collected 3844612 microblogs sent by 115886 users from Sept. 2009 to Jan. 2010. We prepared a medium microblog set (100 thousand tweets) for effectiveness experiment and a large microblog set (1 million tweets) for efficiency experiment.

5.1.2 Data preprocessing

In order to save space or to speed up sampling, the punctuation and stop words (function words, such as *the*, *is*, *at*) in original microblog dataset must be removed before experiments, which appear with high frequency but cannot help topic mining for microblogs. We finished this preprocessing work by using a punctuation list and a stop words dictionary.

5.1.3 Experiment environment

The experiments were conducted on a cluster of 12 nodes (Master node: RAM 8G; CPU Intel Core 2 Duo 3.00GHz, Hard Disk 160G 7200RPM; Slave node: RAM 4G; CPU Intel Pentium 4 3.00GHz; Hard Disk 1.5T 7200RPM; Switch bandwidth: 1G), with Operating System of Ubuntu Linux 9.10 and Hadoop 0.20.2 as MapReduce implementation.

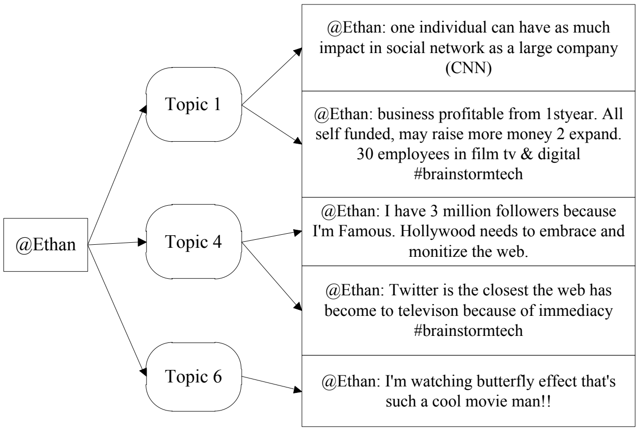
5.2 Effectiveness experiment

Effectiveness experiment was conducted on medium dataset mainly to examine the performance of MB-LDA. The hyperparameters setup refers to [12], where $\alpha = \alpha_d = \alpha_c = 1$, $\beta = 0.01$, $T = 50$. Set $\lambda = 1$ by default to indicate the comment part and original part in retweet microblogs share the same topics.

5.2.1 Performance

The overall result of MB-LDA is showed in Fig. 8, where the first six topics are listed out of total 50 topics. Topics are represented by key words, so we can find out that *Topic 1* is about business topics; *Topic 2* is about Apple's product topics; *Topic 3* is about the event of time; *Topic 4* is about microblog related topics; *Topic 5* is about the event of dining or eating; *Topic 6* is about multimedia topics. The key words of each topic are accurate to recognize and these topics are independent with each other. Fig. 8 also shows the corresponding microblogs of *Topic 2* and *Topic 6*, which are really reasonable in topic assignments. (E.g. the fourth microblog is assigned to *Topic 2* through document relevance relation)

Besides content mining of microblogs, MB-LDA also discovers the topics focused by contactors. According to contactor relevance relation, Fig. 7 shows that Ethan (pseudonym for privacy) pays close attention to *Topic 1, Topic 4* and *Topic 6*. The corresponding microblogs with Ethan really fall into these topics.



@Ethan: one individual can have as much impact in social network as a large company (CNN)

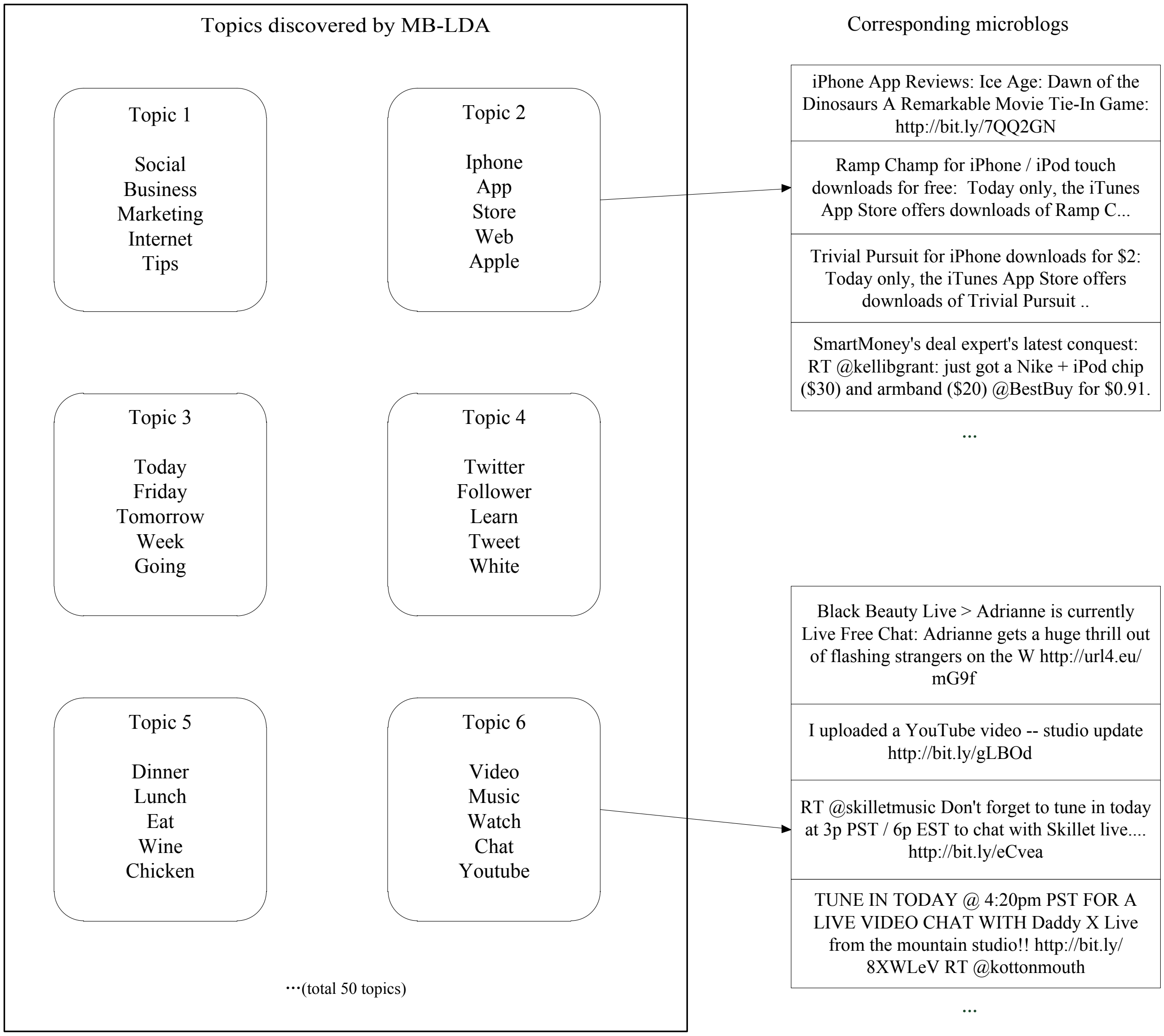
@Ethan: business profitable from 1styear. All self funded, may raise more money 2 expand. 30 employees in film tv & digital #brainstormtech

@Ethan: I have 3 million followers because I'm Famous. Hollywood needs to embrace and monitize the web.

@Ethan: Twitter is the closest the web has become to televison because of immediacy #brainstormtech

@Ethan: I'm watching butterfly effect that's such a cool movie man!!

Fig. 7 Example of contactor-topic relation



iPhone App Reviews: Ice Age: Dawn of the Dinosaurs A Remarkable Movie Tie-In Game: http://bit.ly/7QQ2GN

Ramp Champ for iPhone / iPod touch downloads for free: Today only, the iTunes App Store offers downloads of Ramp C...

Trivial Pursuit for iPhone downloads for $2: Today only, the iTunes App Store offers downloads of Trivial Pursuit ..

SmartMoney's deal expert's latest conquest: RT @kellibgrant: just got a Nike + iPod chip ($30) and armband ($20) @BestBuy for $0.91.

Black Beauty Live > Adrianne is currently Live Free Chat: Adrianne gets a huge thrill out of flashing strangers on the W http://url4.eu/mG9f

I uploaded a YouTube video -- studio update http://bit.ly/gLBOd

RT @skilletmusic Don't forget to tune in today at 3p PST / 6p EST to chat with Skillet live.... http://bit.ly/eCvea

TUNE IN TODAY @ 4:20pm PST FOR A LIVE VIDEO CHAT WITH Daddy X Live from the mountain studio!! http://bit.ly/8XWLeV RT @kottonmouth

Fig. 8 Topic mining overall result of MB-LDA

5.2.2 Comparative experiment

We also conducted the comparative experiment between MB-LDA and LDA, which is a baseline model in the field of topic modeling. The metric Perplexity is a standard measure of performance for statistical models, which indicates the uncertainty in predicting a single word; the model with lower value is better in performance. Perplexity is defined as follows:

$$Perplexity(W) = \exp\left\{-\frac{\sum_m \log p(w_m)}{\sum_m N_m}\right\} \quad (9)$$

where W is a test set with m documents, w_m and N_m indicate the observed words and number of words in the test document respectively.

Perplexity is used to measure the performance of LDA and MB-LDA under the same hyperparameters setup, and the result is showed in Fig. 9:

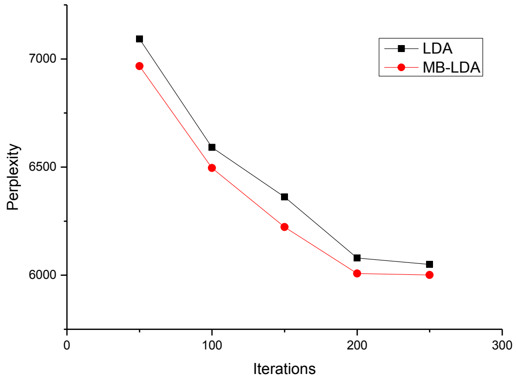


Fig. 9 Perplexity of LDA and MB-LDA

Fig. 9 shows that perplexity of MB-LDA is always lower until they reach the convergence after enough iterations, and confirms that MB-LDA outperforms the baseline of LDA with the help of structured information in microblog itself.

5.3 Efficiency experiment

Efficiency experiment was conducted on large dataset to examine the scalability of distributed MB-LDA. When faced large scale date, MB-LDA on single node becomes less practical or even can’t conduct calculation within limited resources. At that time, distributed MB-LDA offers a solution to handle large scale microblogs.

We measured and compared the speedup of distributed MB-LDA with default setup and with optimized setup to verify the scalability using the large microblog set. Because MB-LDA on single node fails with “out of memory” error, we used 3 nodes as the baseline to measure the speedup of 3/6/9/12 nodes in the efficiency experiment. To quantify speedup, we made an assumption that the speedup of using 3 nodes is 3 compared to using one node.

In our cluster environment, parameters are listed as follows: M=12, R=12 (each slave node with 1 mapper and 1 reducer), V_n=10.8M/s, C_1=3.3s, C_2=3.2s and S'≈6600M. Under these parameters configuration, we can calculate out the optimized setup for X mappers and Y reducers, compared with default setup (M mappers and one reducer). The speedup of using different nodes is showed in Fig. 10.

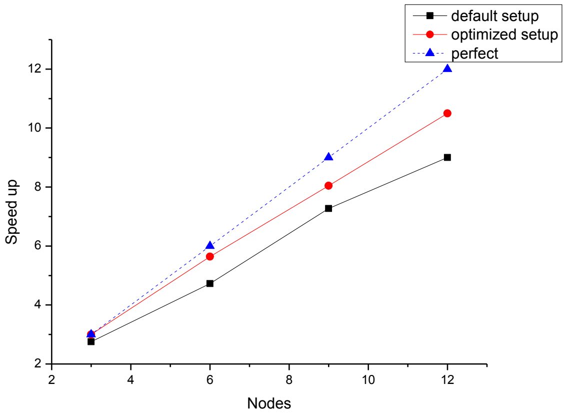


Fig. 10 Speedup of distributed MB-LDA

Fig. 10 shows that distributed MB-LDA can achieve increasing speedup when more nodes are added into the system, although there lies a gap between perfect linear speedup (dash line) and distributed MB-LDA speedup, which is expected due to the increase time spending in network communication over the cluster. Besides, Fig. 10 shows distributed MB-LDA with optimized setup achieves higher speedup than that with default setup, and confirms that the optimization can improve the overall performance. From the view of time (see Table 1), the optimization can reduce the execution time by about 10~15%.

Table 1 Average execution time per iteration

Nodes	Default setup	Optimized setup	Time saving
3	137mins	126mins	8.0%
6	80mins	67mins	16.2%
9	52mins	47mins	9.6%
12	42mins	36mins	14.3%

Furthermore, we analyzed the relationship between sampling time and total map time. We figure out sampling time takes up 86.2% of total map time by average. The remaining time is spent on I/O transfer and network communication. Effort on reducing this time can further improve the efficiency of distributed MB-LDA.

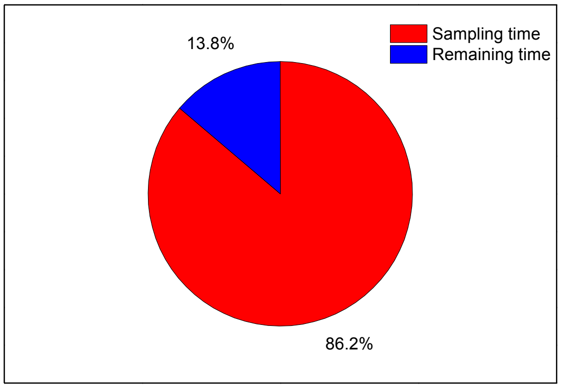


Fig. 11 Time distribution graph of Map step

6. CONCLUSION

In this paper, we introduced MB-LDA for topic mining in microblogs and proposed distributed MB-LDA to handle the situation of large scale microblogs. Experimental results show that MB-LDA exhibits good performance and distributed MB-LDA scales well on actual dataset.

In future work, the automatic learning of hyperparameters in MB-LDA can be conducted to achieve a better performance in various application scenarios. Besides, we plan to use distributed cache to reduce the time spent on I/O transfer and speed up the execution time of distributed MB-LDA. Other strategies or implementations to reduce network communication should be also considered.

7. ACKNOWLEDGMENTS

The work is supported by Ministry of Industry and Information Technology of China (No. 2010ZX01042-002-003-001)

8. REFERENCES

[1] J. H. Kang, K. Lerman, A. Plangprasopchok. Analyzing Microblogs with Affinity Propagation. In *Proceedings of the 1st KDD workshop on Social Media Analytic*, 2010: 67-70

[2] A. Java, X. Song, T. Finin, et al. Why we Twitter: Understanding Microblogging Usage and Communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis (WebKDD/SNA-KDD)* 2007:56-65

[3] B. Krishnamurthy, P. Gill, M. Arlitt. A few chirps about Twitter. In *Proceedings of the first workshop on online social networks (WOSP)*, 2008:19-24

[4] D. Ramage, S. Dumais, D. Liebling. Characterizing microblogs with topic models. In *Proceedings of International AAAI Conference on Weblogs and Social Media*, 2010: 130-137

[5] J. Dean, S. Ghemawat. MapReduce: simplified data processing on large clusters. *Commun*. 2008, 51(1): 107-113

[6] R. Xu, D. Wunsch. Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 2005, 16(3): 645–678

[7] S. Deerwester, S. Dumais, T. Landauer, et al. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 1990, 41(6): 391–407

[8] G. Salton, M. McGill. *Introduction to Modern Information Retrieval*. New York: McGraw-Hill, 1983

[9] T. K. Landauer, P. W. Foltz, D. Laham. Introduction to Latent Semantic Analysis. *Discourse Processes*, 1998, 25: 259-284

[10] D. M. Blei, A. Y. Ng, M. I. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 2003, 3: 993–1022

[11] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual International ACM SIGIR Conference on Research and development in information retrieval*, 1999: 50-57

[12] T. Griffiths, M. Steyvers. Probabilistic topic models. *Latent Semantic Analysis: A Road to Meaning*. Hillsdale, NJ: Laurence Erlbaum, 2006

[13] X. Wei and W. B. Croft. LDA-based document models for ad hoc retrieval. In *Proceedings of the 29th annual International ACM SIGIR Conference on Research and development in information retrieval*, 2006: 178-185

[14] L. Dietz, S. Bickel, T. Scheffer. Unsupervised prediction of citation influences. In *Proceedings of the 24th International Conference on Machine learning*, 2007: 233-240

[15] QiaoZhu Mei, Deng Cai, Duo Zhang, et al. Topic Modeling with Network Regularization. In *Proceedings of the 17th International Conference on World Wide Web*. 2008

[16] D. M. Blei, J. Lafferty. Topic models. *Text Mining: Classification, Clustering, and Applications*. New York: Chapman & Hall/CRC, 2009

[17] R. Nallapati, W. Cohen. Link-pLSA-LDA: A new unsupervised model for topics and influence of blogs. In *Proceedings of the International Conference on Weblogs and Social Media (ICWSM)*, 2008

[18] Congkai Sun, Bin Gao, Zhenfu Cao, et al. HTM: A topic model for hypertexts. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, 2008: 514-522

[19] D. Newman, A. Asuncion, P. Smyth, et al. Distributed Algorithms for Topic Models. *Journal of Machine Learning Research*. 2009, 1801-1828.

[20] A. Asuncion, P. Smyth, M. Welling. Asynchronous distributed learning of topic models. In *Proceedings of the 20th Neural Information Processing Systems (NIPS)*. 2008

[21] Yi Wang, Hongjie Bai, M. Stanton, et al. PLDA: Parallel Latent Dirichlet Allocation for Large-Scale Applications. In *Proceedings of the 5th International Conference on Algorithmic Aspects in Information and Management (AAIM '09)*, 2009, 301-314.

[22] A. Smola, S. Narayanamurthy. An architecture for parallel topic models. *Proceedings of VLDB Endow*.2010, 3, 1-2, 703-710.

[23] T. L. Griffiths, M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 2004, 101:5228–5235,

[24] T. P. Minka, J. Lafferty. Expectation-propagation for the generative aspect model. In *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence*, 2002: 352-359

[25] Xiao Yang, Jianling Sun. An Analytical Performance Model of MapReduce. In *Proceedings of CCIS*: 306 – 310, 2011.