Recommender Systems

Definition

Types: 1) Collaborative, Content-Based, Hybrid

NOTES:

Write ABOUT LDA itself -- > Probabilistic Topic Models of CTR paper

MN - my new addition

OPP+MN – other papers paraphrase + my new addition

PP – paper paraphrase

PO- paper original

Hybrid Methods

These methods, that combine collaborative filtering and content based approaches, achieves state-of-the-art results in many cases and are, so, used in many large scale recommender systems nowadays. Hybrid recommender systems benefit from their (cf and cb) complementary advantages.

CTMP with BOPE

MN

In this paper, we investigate the application of Bernoulli randomness for Online Maximum a Posteriori Estimation (BOPE) algorithm for solving MAP problem in Collaborative Topic Model for Poisson distributed ratings (CTMP) which is a hybrid model for recommender systems. We also show that performance behaviour of CTMP is better with BOPE compared to OPE which was used in original paper where CTMP was proposed.

Abstract-PP

CTMP is a hybrid and interpretable probabilistic content-based collaborative filtering model. It allows both content representation by admixture topic modelling and computational efficiency via Poisson factorization coexisting together within a tightly linked probabilistic model, and therefore overcoming a limitation of existing models. CTMP's predictive ability is particularly good in different real-world recommendation contexts, and it is able to scale to very large datasets along with maintaining interpretable user profiles. Another crucial advantage of CTMP is that its sparse representation enables efficient storage of item contents which makes it efficient and desirable model in terms of industrial settings. Therefore, CTMP has been used successfully in industry due to its mentioned benefits.

Introduction-PP

CTMP, being a probabilistic hybrid model with scalability and interpretability, offers the following contributions:

1)

2)

3)

Related Work in Comparison with CTMP – PP

There has been a lot of interest in combining modelling content with matrix factorization in the field of hybrid recommender systems. Especially, by representing item content with topic models [\*\*\*\*], the models can also benefit from the interpretable semantics of the latent space characterized by the topic mixtures, which makes the semantics of item latent factor more interpretable [\*\*ctmp\*\*].

Initially, Agarwal and Chen proposed topic models in matrix factorization with fLDA [\*\*\*] where the item latent factor takes the role of topic proportion in the LDA representation. Despite of their accurate and interpretable model which handles both cold-start and warm-start scenarios, it still has a limitation in distinguish items which have an identical topic mixture, but content details that topic mixture can not cover are of concern to different groups of people.

To explain it further, consider that we have two articles, namely, A and B, and both of them are about application of machine learning to social networks. Because they are same in terms of their contents, they will also have same topic proportions. Now let’s consider that these two articles are of interest to different kind of users: Article A provides an intriguing machine learning algorithm which is applied to social network applications, wheares article B implements standard machine learning algorithm, but provides a crucial data analysis on social network data [\*\*CTR\*\*]. As a result, users who work in machine learning will prefer article A and will hardly be interested in article B, wheares users who work in social networks will be more interested in article B instead of A. However, as the topic proportions of both articles are same, both of them will be recommended to both groups of users. To tackle this problem, a novel approach called Collaborative Topic Regression (CTR) [\*\*CTR\*\*] has been proposed which addresses the limitation of fLDA by making the item latent factor be an offset from topic proportion. By this way, the model allows the item latent factor to also cover the contribution of user ratings. In result, CTR has demonstrated major improvements in comparison to fLDA.

Despite its advantages, CTR still has a serious computational limitation. The reason behind this is that the model considers user ratings to have a Gaussian distribution which leads to iterating over all of the entries in rating matrix during training. Because of this, CTR is highly inefficient considering that real-world datasets are very big and sparse.

In order to address the inefficiency of CTR, a newer model called Collaborative Topic Poisson Factorization (CTPF) has been proposed which filled in inefficiency gap of CTR by considering that ratings have Poisson distribution. CTPF is only concerned with non-zero ratings during training, and therefore it is much more efficient and scalable. Nonetheless, for the purpose of making sure that the model is conditionally conjugate and has closed-form updates, CTPF tries to model the content generation by standard mixtures of Gamma.

CTMP algorithm which is used in this paper covers the limitation of CTR by considering ratings in Poisson distribution as CTPF does, while modelling contents with LDA [\*\*ctmp\*\*]. CTMP also makes the following contributions to the previous approaches:

1. CMTP has been tested in variety of fields where real-world recommendation is the most challenging one. It has been seen that CTMP outperforms the other existing models significantly. Its main competency is in recommending scientific articles and commercial product recommendation. Recommending movies are also amongst these, and indeed, this is what we will test on this paper.
2. Coordinate ascent algorithm is carried out because CTMP is non-conjugate model. The algorithm is fast and also scalable.
3. [[[[[SPARSITY???? Needs proof]]]

**A Collaborative Topic Model for Poisson distributed ratings**

In this section, we describe learning, prediction phases and key properties of CTMP – hybrid and interpretable probabilistic content-based collaborative filtering model.

ADD BELOW to DATASET DESCRIPTION PART

Graphical user interface, text, application, email

Description automatically generated

CTMP.1 - Formalization

Before going into technical parts, let’s provide some notations:

* *U:* represents the number of users inside the dataset
* *J:* representsthe number of items inside the dasaset
* describes the bag-of-word representation for each item *j* where expresses the frequency of term/word in item *j.*
* represents the vocabulary size of the corpus.
* describes the dataset where is a rating provided by user *u* to item *j,* while is the bag-of-word representation of item *j* as already explained above. represents the ratings given to movies by users. Every rating is expressed as binary 0 or 1. If user *u* liked an item *j,* then . On the contrary, if the user *u* do not know about the item *j* or do not like it, then .
* *K:* represents the number of topics inside corpus.
* describes the topic representation. More precisely, every topic *k* is a distribution over the vocabulary. It is described as and . Note that, lies in the (*k* – 1)-simplex.
* describes the topic proportion of the items. is the vector of the distribution on topics for item *j,* and . Note that, lies in the (*k* – 1)-simplex.

In order to learn the topics , we use the Latent Dirichlet allocation (LDA) and its Expectation-Maximization (EM) approach which was described in the respective section of LDA. Furthermore, by learning the topic proportion of each item we describe each item and user in the *K*-dimensional space. Note that these learning procedures will be explained in the further sections below.

Now, we present **latent factors** for each user and item in terms of *K*-dimensional vectors and , respectively. As discussed in [[[Related Work Section]]], the reason why we consider rather than as the latent factor for item is that in order to have better recommendation system, we allowed an offset between and which accounts for the user-specific preference on the item content that alone can not capture. Therefore we denote that where is an offset term which has Gaussian distribution. Note that in the formula above represents an *K*-dimensional identity matrix , and is a regularization parameter. So, we have .

Furthermore, as shown below, the ratings and users’ latent factors are modeled by Poisson and Gamma distributions, respectively. To put everything together, the generative process of CTMP is as follow:

Text, letter

Description automatically generated

1. For each user *u,* draw where
2. For each item *j*:
3. Draw topic proportion
4. For the *n*th word of item *j*:
   1. Draw topic index
   2. Draw word
5. Draw latent factor
6. For each user-item pair (*u, j*), draw

Note that step 2(a-b) correspond to LDA.

Below is graphical representation of CTR, CTMP and CTPF:

Diagram, engineering drawing

Description automatically generated

**Learning CTMP**

Full posterior of latent variables is given as follow:

(1)

The problem with this posterior is that it is intractable, and therefore exact inference is impossible. In order to tackle this problem, we have two methods:

1. Maximum A Posteriori (MAP) for point estimation
2. Bayesian Learning such as MCMC Sampling or Variational Methods for approximate inference

As the prior and posterior distributions of hidden variables and are not conjugate in CTMP model, using Variational Methods of Bayesian Learning in order to infer these hidden variables does not get us closed-form solution. Therefore, we will carry out the point estimates of and using MAP – coordinate ascent algorithm developed by authors of original paper of CTMP [\*\*CTMP\*\*].

Furthermore, in order to facilitate the learning, authors added **a** **new auxiliary variable *y****,* where Poisson( and . Note that we approximate the posterior of and via mean-field variational inference [\*\*5\*\*]. In result, we convert the inference problem of CTMP into a full optimization problem where the single objective function which needs to be maximized is as follow:

(2)

As shown in (2), the term of integration and summation over the whole space causes optimization to be intractable. However, Variational method [\*\*5\*\*] also tackles this problem which will be discussed in detail below.

Note that has Poission distribution, and the *K*-dimensional vector follows multinomial distribution: [\*\*6\*\*] So, we get the variational distribution as follow:

(3)

where such that is variational parameter of , and are variational parameters of . Note that . Now we get the **evidence lower bound (*l*)** by applying Jensen’s inequality[[\*\*insertpaperofJensen\*\*]]:

(4)

Note that before learning the hidden parameters, are considered as fixed parameters in the model.

Next, the lower bound *l*( shp, rte, ) is maximized with respect to shp, rte, . According to Appendix A, we express the terms in detail as follow:

(5)

**Learning Parameters**

Equation (5) is the optimization problem and as mentioned before we solve it by coordinate ascent algorithm. We summarize the whole learning in Algo-rithm1. The update expressions of parameters θ, μ, φ, shp, rte, βare described as follow.

**Learning**

.

.

.

.

.

.

**LDA**

In machine learning, topic modeling is a statistical model for discovering the abstract topics that occur in a collection of documents. [1] It is also considered as a probabilistic model for an unsupervised classification of the documents.

There are many topic modeling algorithms, among which Latent Dirichlet allocation (LDA) is the most popular one. LDA is a three-level hierarchical Bayesian model, and its basic idea is that documents are represented as random mixtures over underlying set of topics, where each topic is characterized by a distribution over words. [\*\*original-lda-paper\*\*]. By this way, topic probabilities express an explicit representation of each document. This can also be be explained as:

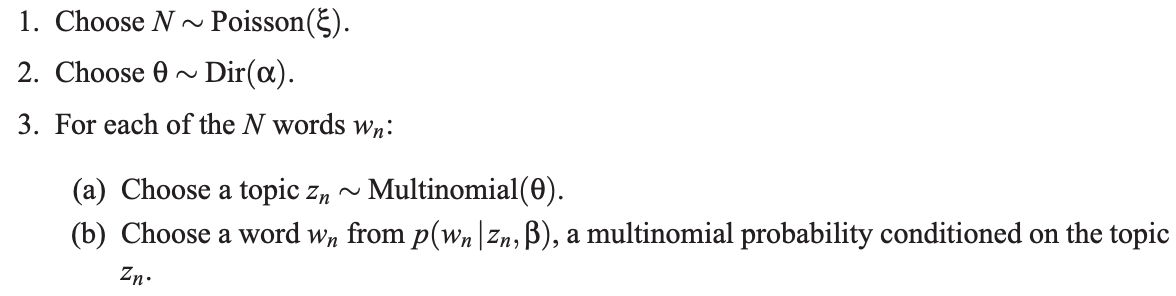
* **Each document is a mixture of topics**. By this way, we consider that each document contains terms/words from some topics in specific proportions. For instance, if we consider that there are 2 topics in the whole corpus, then we might say some document could be 75% topic A, and 25% topic B, while another document might be consisted of 30% topic A, and 70% topic B”.
* **Each topic is a mixture of words**. In this case, we consider that each topic is expressed by the words that explain it most. For example, if we consider that there are 2 topics, namely, “sports” and “education”, in the whole corpus, then the most used words for the sports topic could be “teammate”, “win”, and “play”, while the education topic could contain the words such as “lecture“, “book” and “class”. It is necessary to note that the same words can appear on the multiple topics. For example, the word “time” could participate in both sports and education topics.

By this way, documents overlap with each other with relationship to their contents, rather than being seperated into different individual groups.

To put it into terminology:

* A *word* is a term of the vocabulary and it is indexed by
* A *document* is a series of words given by where is the *n*th word inside the document.
* A *corpus* is a collection of a total *M* documents and it is given by

Generative process of LDA for each document in the whole corpus is described below:



In the figure above, is the topic proportion of some document, is the topic representation of some topic and is the Dirichlet prior parameter. The Poisson assumption of the proccess is not essential to all that follows and more realistic document length distributions can be used as appropriate. So, *N* is completely independent of all data generating variables such as z. [\*\*original-lda\*\*]

Topic proportion is a *k*-dimensional Dirichlet random variable and its domain is in the (*k* − 1)-simplex. In other words, *k*-vector is in the (*k* − 1)-simplex if . The Dirichlet is a exponential family distribution on the simplex. Some of its important properties are that it is conjugate to the multinomial distribution. [\*\*4\*\*]. Note that, all of these properties help the development of the inference and parameter estimation for LDA. Additionally, the probability density of ’s simplex is as following:

,

where is a *k*-vector with components and is the Gamma function.

Probabilistic graphical model of LDA is represented below [\*\*original-lda\*\*]:

Diagram, schematic

Description automatically generated

The outer box illustrates the documents, and the inner box illustrates repeated choice of topics and words inside documents [\*\*original-lda\*\*]. According to the figure above it is once again shown that LDA is a three-level hierarchical Bayesian model. The parameters that are considered as corpus-level areand , and they are supposed to be sampled during the process of generating the corpus. [\*\*original-lda\*\*]. The variables given as are considered as document-level, and they are sampled once per document. Lastly, the variable given as and are considered as word-level, and they are sampled once for each word inside every document. During the LDA process, topics are sampled repeatedly within each document.

So, the joint distribution of topic proportions topics set of topics z, and set of words w is as following:

[[[[[because above posterior distribution is intractable, we can use variational inference]]]]

[[[[[ geometric interpretation of LDA]]]]

[[[[ Learning theta, beta by EM algorithm because EM was referenced in Formalization part of CTMP]]]

Geometric interpretation of LDA

Illustrating the geometry of the latent space is a another good way for grasping the concept of LDA. As shown below in the figure,

**Inference and Parameter Estimation**

Computing the posterior distribution over the hidden variables given some document is the the main problem we have to overcome in order to be able to use LDA:

Unfortunately, the posterior distribution derived above is intractable for an exact inference. Fortunately, there are some methods of approximate inference which can be used for LDA.

One of them is Variational Approximations which facilitate approximate inference for the parameters in complex statistical models. The main idea behind the variational inference is to get an adjustable lower bound on the log likelihood by using Jensen’s inequality (Jordan et al., 1999). ~~For this reason, we use free variational parameters which try to find the tightest possible lower bound.~~ [[[[[ABOUT VARIATIONAL INFERENCE]]]]]]

In order to be able to deploy variational inference method for getting the tractable family of lower bounds, we make some adjustments to our original graphical model of LDA:

**Diagram

Description automatically generated**

Where respectively.

Now, our aim is to optimize the problem that determines the values of the variational parameters and . This is exactly what finding a tight lower bound on the log likelihood is.

APPENDIX A

.

.

.

.

.

.

[1] <https://en.wikipedia.org/wiki/Topic_model>

2?? <https://www.tidytextmining.com/topicmodeling.html>

[3] <https://dl.acm.org/doi/pdf/10.5555/944919.944937>

[4] <https://stephentu.github.io/writeups/dirichlet-conjugate-prior.pdf>

[5] <https://people.eecs.berkeley.edu/~jordan/papers/variational-intro.pdf>

[6] Univariate Discrete Distributions, vol. 444 [[[[take from CTMP]]]]

https://people.eecs.berkeley.edu/~jordan/papers/variational-intro.pdf -

In particular, they make a link between this lower bound and parameter estimation via the EM algorithm