A Topic Model Based on Poisson Decomposition

Haixin Jiang[†¶], Rui Zhou[§], Limeng Zhang[†¶], Hua Wang[‡], Yanchun Zhang[‡¶]

[†]School of Computer and Control Engineering, University of Chinese Academy of Sciences, China
[§] Department of Computer Science and Software Engineering, Swinburne University of Technology, Australia
[‡]Centre for Applied Informatics, Victoria University, Australia
[¶]School of Computer Science, Fudan University, China
[†]{jianghaixin13b, zhanglimeng13}@mails.ucas.ac.cn
[§]rzhou@swin.edu.au
[‡]{hua.wang, yanchun.zhang}@vu.edu.au

ABSTRACT

Determining appropriate statistical distributions for modeling text corpora is important for accurate estimation of numerical characteristics. Based on the validity of the test on a claim that the data conforms to Poisson distribution we propose Poisson decomposition model (PDM), a statistical model for modeling count data of text corpora, which can straightly capture each document's multidimensional numerical characteristics on topics. In PDM, each topic is represented as a parameter vector with multidimensional Poisson distribution, which can be easily normalized to multinomial term probabilities and each document is represented as measurements on topics and thereby reduced to a measurement vector on topics. We use gradient descent methods and sampling algorithm for parameter estimation. We carry out extensive experiments on the topics produced by our models. The results demonstrate our approach can extract more coherent topics and is competitive in document clustering by using the PDM-based features, compared to PLSI and LDA.

CCS CONCEPTS

• **Information systems → Document topic models**; **Content analysis and feature selection**;

KEYWORDS

Topic model, Poisson decomposition, statistical testing, text classification, topic coherence

1 INTRODUCTION

Topic modeling is an effective tool for uncovering the underlying semantic structure of text corpora and other collections of count data [3, 7, 41, 45]. With this tool we can automatically find useful patterns so as to explore and organize large collections of electronic archives. It has been applied to many kinds of documents, including scientific abstracts [4, 20] and newspaper archives [45]. Topic models have also served as a powerful technique to discover patterns of words and semantic structure in otherwise unstructured collections in different fields, such as natural language processing [5, 21, 42–44], opinion mining [6, 29–31, 33], information retrieval [12, 40, 45, 46], topic segmentation [7, 25] and collaborative filtering [13, 14, 23, 27].

In topic modeling area, the basic methodology for text corpora reduces each document in the corpus to a vector of counts of terms. When analyzing text, these patterns, which are often called "topics", are represented as distributions over words and often placed high probability on semantically meaningful sets. Deerwester et al.[16] proposed Latent semantic indexing (LSI) to reduce document description length and to reveal document statistics structure. LSI used a singular decomposition of the matrix to identify a linear subspace and achieve significant compression in large collections. As an alternative to LSI, the probabilistic LSI (PLSI) [22], a generative probabilistic model, models each word in a document as a sample from a mixture model where each word is generated from a single topic and different words are from different topics. However, Hofmann's work provides no probabilistic model at the level of document. Based on the seminal works in LSI and PLSI, Blei et al. proposed Latent Dirichlet allocation (LDA) [4], in which each word of a collection is modeled as a finite mixture over an underlying set of topics and each topic is modeled as an infinite mixture over an underlying set of topic probabilities.

In the above models, word proportions in a topic and topic proportions in a document are the goals to learn and they are used to represent a topic and a document separately. However, word proportions, as well as topic proportions are far from enough to a topic and to a document. Counts vectors imply more than word proportions. Addressing this gap, we propose a statistics-based model for count data, which uses Poisson distribution to model occurrence of topics in each document. Counts imply more important statistic properties and this demand brings forth the notion of the so called Poisson decomposition model(PDM). Here we use the same word "decomposition" as Raikov.D did in [38]. We believe that "decomposition" instead of "factorization" can better reflect Theorem 4.2

CIKM'17 , November 6–10, 2017, Singapore, Singapore
© 2017 Association for Computing Machinery.
ACM ISBN 978-1-4503-4918-5/17/11…$15.00
https://doi.org/10.1145/3132847.3132942

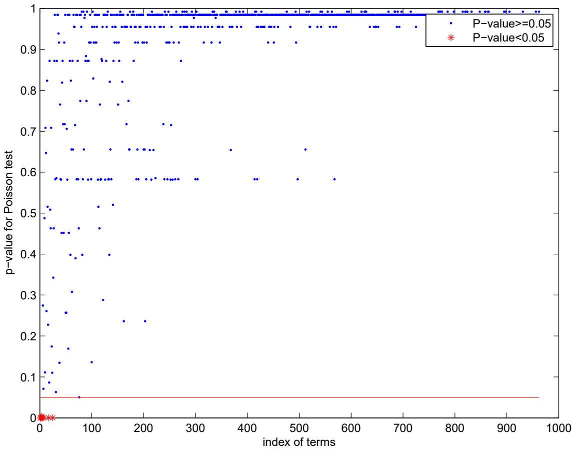

Figure 1: Hypothesis Testing for Poisson distribution on the NIPS datasets. Similar testing results also appear in the Reuters-21578, TDT2 and 20Newsgroups datasets.

and also differ from general "Poisson factorization" for recommendation [10, 11, 18]. PDM frees the restriction on word proportions in a topic and topic proportions in a document and is able to obtain their numerical characteristics of documents, which are the biggest difference between PDM with other topic models.

Like the models above, PDM is also based on the exchangeability assumption for the words and for the documents, in which the orders of words in a document and the orders of documents in a corpus can be neglected [1]. According to de Finetti[15], any collection of exchangeable random variables has a representation as a mixture distribution. Therefore, PDM considers about the exchangeability of words in documents and represents each document as a mixture distribution.

We claim that using Poisson distributions to model count data is proper for accurate estimation of structure characteristics. We perform a hypothesis test in statistics to test the validity of a claim and p-values help us determine the significance of statistical distributions for modeling count data. In our hypothesis test, when the p value is less than or equal to 5% ($p \leq 0.05$), we reject the Poisson hypothesis. We will refer to $p < 0.05$ as the criterion for deciding to reject the Poisson hypothesis, although note that when $p = 0.05$, the decision is also to reject the Poisson hypothesis. When the p value is greater than 5% ($p > 0.05$), we retain the Poisson hypothesis. Figure 1 shows the results of hypothesis testing. It should be pointed out that the Poisson hypothesis is tested on the assumption that the count data comes from unique topic or feature because at this moment no other topics have been obtained. Only after topic model is used to the count data and each word is allocated to some topic can we fit Poisson Hypothesis to test the counts of each topic in all documents.

In this paper we consider the count data of a corpus (a collection of documents). Count data is a statistical data type, a type of data in which the observations can take only the non-negative integer values {0, 1, 2, 3, ...}, and where these integers arise from counting rather than ranking. The contributions of this paper are three-fold: 1) We demonstrate the rationality of Poisson distributions for count data and propose a new statistics topic mode. 2) We develop a sampling method to infer the parameters for our model. 3) We evaluate our models using topic coherence measurement and document clustering performance.

Table 1: Notations

Symbols	Definitions
V	vocabulary and vocabulary size
D	the collection of documents
M	the number of documents in D
N_m	the total word number in document m
K	a specified number of topics
x_{mv}	times that word v occurs in document m
w_{mk}	times that topic k occurs in document m
θ_{mk}	parameters for topic k in document m
η_{kv}	parameters for w_{mk}

The paper is organized as follows. In Section 2 and 3, we introduce basic notation and terminology and also related work. We test the validity of a claim that the count data has a Poisson distribution and present preliminaries on related distributions in Section 4. Our models and parameter estimation are described in Section 5. In section 6, we present topic coherence and document clustering experiments that compare the performance with PLSI and LDA based methods. Finally, we present our conclusions in Section 7.

2 NOTATION AND TERMINOLOGY

We use the language of text collections throughout the paper, referring to entities such as 'words', 'documents', and 'corpora' and define the following terms: a word is the basic unit of discrete data, defined to be an item from a vocabulary; a document is a collection of N words denoted by $\mathbf{w} = (w_1, ..., w_N)$, where w_n is the n^{th} word in the sequence. Detailed notations and terminology are shown on Table 1. A basic vocabulary of 'words' is chosen, and for each document in the corpus, a count is formed by the number of occurrences of each word. Therefore, a collection could be denoted a term-by-document matrix X whose columns contain the count values for each document in the corpus. We also define special notations and terminology for our model. In this paper, the k^{th} topic is defined by $\lambda_k = (\lambda_{k,1}, ...\lambda_{k,V}), k = 1, ..., K$, where $\lambda_{k,v}$ is a Poisson parameter of word v in topic k, and the m^{th} document is defined by $D_m = (D_{m,1}, ...D_{m,K}), m = 1, ..., M$, where $D_{m,k}$ means how many times topic k has been chosen in document m, that is, $D_{m,k}$ is integer. With theorem 4.3 $\lambda_{k,n}$ also means the importance of word n in topic k.

3 RELATED WORK

There have been many papers building on Poisson decomposition (or factorization) to model discrete data [8, 10, 11, 17, 19, 39]. Canny J [10] chose Poisson distribution and Gamma distribution to model document corpus, 'because they gave a very clean and simple algorithm which is highly efficient.' Gopalan P et al. [19] assumed each observation is drawn from a Poisson distribution, whose rate is a linear combination of the corresponding user and item attributes.

Gopalan P et al. [17] used Poisson distribution to model the observations for recommendation systems, parameterized by the inner product of the user preferences and item attributes. Charlin L et al. [11] extended Poisson factorization to handle time series of clicks. Buntine W [8] discussed mainly multinomial PCA and mentioned only the relation between Poisson variables and multinomial. To sum up, these works above employed directly Poisson factorization and pointed out the additivity of independent Poissons. What the models factorize are components or preferences. However, they did not give detailed reasons for choosing Poisson factorization and did not fully utilize the decomposition properties. Different to the works above where Poisson decomposition is employed directly to model discrete observations, this paper justifies the application of decomposition properties in the model and builds a solid theoretical basis for Poisson decomposition topic models on word counts and topic counts. Here, different from the existing models, what this model factorizes are variables. To the best of our knowledge, we are the first to describe the importance of Raikov's theorem of Poisson distribution when modeling discrete data. Especially, we give detailed theoretical analysis. Another novelty is that we develop constrained optimization topic models and substitute sampling methods for variational EM to compute these models.

4 PRELIMINARIES ON RELATED DISTRIBUTIONS

In this section, we review some related distributions and their properties.

4.1 The Poisson Distribution

We start with the Poisson assumption. A discrete random variable X is said to have a Poisson distribution with parameter $\lambda > 0$, if, for $k = 0, 1, 2, \ldots$, the probability mass function of X is given by

$$P(X = k) = \frac{\lambda^k}{k!} e^{-\lambda},$$

where e is the Euler's number and $k!$ is the factorial of k. The parameter λ is equal to the expected value of X and also to its variance.

The Poisson distribution has very nice properties that we will use in our model. It arises as the distribution of counts of occurrences of events in multidimensional intervals in multidimensional Poisson processes in a directly equivalent way to the result for unidimensional processes. Thus, if D is any region of multidimensional space for which $|D|$, a kind of measurement of the region, is finite, and if $N(D)$ is count of the number of some event in D, then

$$P(N(D) = k) = \frac{(\lambda|D|)^k e^{-\lambda|D|}}{k!}.$$

4.2 Auxiliary Theorems

We illustrate how to use Poisson distribution to model the statistical numeric characteristics for a corpora. Here we focus on properties of Poisson distribution and the close relations between Poisson distribution and Multinomial distribution.

Theorem 4.1 (Sum of Poisson Random Variables). *If X_1 and X_2 are independent Poisson random variables where X_i has a Poisson (λ_i) distribution for $i = 1, 2$, then $X_1 + X_2$ has a Poisson distribution with $\lambda_1 + \lambda_2$. More generally, if X_1 , ..., X_k are independent Poisson random variables where X_i has a Poisson (λ_i) distribution for $i = 1, 2, ..., k$, then $X_1 + ... + X_k$ has a Poisson$(\lambda_1 + \lambda_2 + ... + \lambda_k)$ distribution.*

According to additive properties of independent Poisson random variables [28], given that $x_k, k = 1 \ldots K$ are independent variables with $x_k \sim Poisson(\lambda_i)$, their sum $x = \sum_k x_k$ follows $Poisson(\sum_k \lambda_k)$. A converse is Raikov's theorem, which says that if the sum of two independent random variables is Poisson-distributed, then so is each of those two independent random variables. The following decomposition theorem was derived by D. A. Raikov in [38].

Theorem 4.2 (Raikov's theorem,[32, 38]). *If the sum of two independent nonnegative random variables X_1 and X_2 has a Poisson distribution, then both X_1 and X_2 themselves must have Poisson distributions. It can readily be shown by mathematical induction that the same is true for the sum of more than two independent random variables.*

Raikov's theorem implies that if a Poisson random variable X can be decomposed as $X = X_1 + X_2$, where $X_1 \geq 0$ and $X_2 \geq 0$ are independent random variables (neither of which is identically zero), then both X_1 and X_2 are Poisson. It can easily be shown by mathematical induction that the same is true for the sum of than more than two independent random variables.

Theorem 4.3. *If X_1 and X_2 be independent Poisson random variables where X_i has a Poisson (λ_i) distribution for $i = 1, 2$, then the distribution of X_1 conditional on $X_1 + X_2$ is a binomial distribution. More generally, if X_1, ..., X_k be independent Poisson random variables where X_i has a Poisson (λ_i) distribution for $i = 1, 2, ..., k$, given $\sum X_k = N$ then $(X_1, ..., X_k)$ has a Multinomial (N, π) distribution, where $\pi = (\frac{\lambda_1}{\sum \lambda_k}, ..., \frac{\lambda_k}{\sum \lambda_k})$.*

This fact is important, because it implies that the unconditional distribution of $(X_1, ..., X_k)$ can be factored into the product of two distributions: a Poisson distribution for the overall total, $N \sim P(\lambda_1, ..., \lambda_k)$ and a multinomial distribution for $X = (X_1, ..., X_k)$ given N, $X \sim Multi(N, \pi)$. In this case, it's reasonable to regard the $X_1, ..., X_k$ as independent Poisson random variables with means $\lambda_1, ..., \lambda_k$. If our interest lies not in the λ_k's but in the proportions of variables, inferences about these proportions will be the same whether we regard the sample size N as random or fixed. That is, we can proceed as if $X = (X_1, ..., X_k) \sim Multi(N, \pi)$ even though N is actually random.

5 POISSON DECOMPOSITION MODELS

Several simplifying assumptions are made in this basic model, some of which we remove in subsequent sections. First, the dimensionality K of multivariate Poisson distribution (and thus the dimensionality K of topic presentation of each document) is assumed known and fixed. Second, the words for each topic are parameterized by an N-vector $(\lambda_1, \ldots, \lambda_N)$, which we treat as a fixed quantity to be estimated. Finally, count measurements D_{mk} of topics in a document, which are not considered as parameters but variables, are integers. We should also note that D_{mk} is count measurements for topic k to be chosen in document m and it differs from the number of words allocated to topic k in document m, that is, $\sum_k D_{mk} \neq N_m$; nevertheless, they are the same when $(\lambda_1, \ldots, \lambda_N)$ sum to one. This is because $N_m = E(\sum_k D_{mk}) = \sum_n \sum_k D_{mk}\lambda_{kn} = \sum_k D_{mk} \sum_n \lambda_{kn}$ $= \sum_k D_{mk}(\sum_n \lambda_{kn}) = \sum_k D_{mk}$.

Word counts matrix of document d is denoted as

$$Y_m = \begin{bmatrix} Y_{m11} & \ldots & Y_{m1V} \\ \ldots & Y_{mkv} & \ldots \\ Y_{mK1} & \ldots & Y_{mKV} \end{bmatrix}, \tag{1}$$

where Y_{mkv} is word counts variable, denoting how many times the v^{th} word occurs because of the k^{th} topic in the m^{th} document. We use upper letters Y_{mkv} to denote variables and lowercase letters y_{mkv} observed data.

For the word v in document d,

$$Y_{mkv} \sim \text{Poisson}(\lambda_{mkv}).$$

Y_{mkv} means the times that the word v is assigned to topic k in document m and in hence is jointly determined by document m, topic k and word v. By rewriting λ_{mkv} as $\lambda_m \times \theta_{mk} \times \eta_{kv}$ we obtain

$$Y_{mkv} \sim Poisson(\lambda_m \times \theta_{mk} \times \eta_{kv}),$$

that is,

$$p(Y_{mkv} = y_{mkv}|\theta, \eta) = \frac{(\lambda_m \theta_{mk} \eta_{kv})^{y_{mkv}}}{y_{mkv}!} \exp\{-\lambda_m \theta_{mk} \eta_{kv}\}.$$

The likelihood function for a corpus is

$$p(y|\theta, \eta) = \prod_{m,k,v} \frac{(\lambda_m \theta_{mk} \eta_{kv})^{y_{mkv}}}{y_{mkv}!} \exp\{-\lambda_m \theta_{mk} \eta_{kv}\}. \tag{2}$$

Learning the proposed model can be derived via gradient ascent on a log-likelihood function which directly follows from the probability distribution Poisson(λ_{mkv}).

$$\begin{aligned} L(y|\theta, \eta) = &\sum_m N_m \log \lambda_m + \sum_{m,k,v} (y_{mkv} \log \theta_{mk} \\ &+ y_{mkv} \log \eta_{kv} - \log y_{mkv}! - \lambda_m \eta_{kv} \theta_{mk}). \end{aligned} \tag{3}$$

It should be noted that we know only some numerical characteristics of y , that is, the variables y and the parameters θ and η are unknown. It turns to a constrained optimization model. Our goal is to estimate y, θ and η that maximize the log-likelihood function subject to constraints. We use the method of Lagrange multipliers to convert the constrained problem to an unconstrained problem. We start the derivation by writing the objective function as

$$\begin{aligned} (\hat{y}, \hat{\theta}, \hat{\eta}) =& argmaxL(y|\theta, \eta), \\ s.t. &\sum_k y_{mkv} = x_{mv}, \ \forall\ m,\ v, \\ &\sum_k \theta_{mk} = 1, \ \forall\ m, \\ &\sum_v \eta_{kv} = 1, \ \forall\ k. \end{aligned} \tag{4}$$

5.1 Maximum Likelihood Estimation

Optimization of these parameters can be achieved via the gradient ascent procedure. The derivatives of the likelihood function with respect to each element of the variables y_{mkv} and the parameters θ_{mk}, η_{kv} yield

$$\begin{aligned} \frac{\partial L}{\partial \lambda_m} &= \frac{N_m}{\lambda_m} - 1, \\ \frac{\partial L}{\partial \theta_{mk}} &= \frac{\sum_v y_{mkv}}{\theta_{mk}} - N_m, \\ \frac{\partial L}{\partial \eta_{kv}} &= \frac{\sum_m y_{mkv}}{\eta_{kv}} - \sum_m N_m \theta_{mk}, \\ \frac{\partial L}{\partial y_{mkv}} &= \log \lambda_m \theta_{mk} \eta_{kv} - \Psi(y_{mkv} + 1). \end{aligned} \tag{5}$$

Here $\Psi(x)$ is the digamma function, the logarithmic derivative of the gamma function.. Following gradient method, the resulting update rules for θ and η yield

$$\begin{aligned} \theta_{mk} &= \frac{\sum_v y_{mkv}}{N_m}, \\ \eta_{kv} &= \frac{\sum_m y_{mkv}}{\sum_{mv} y_{mkv}}, \\ y_{mkv} &= \Psi^{-1}(\log N_m \theta_{mk} \eta_{kv}) - 1. \end{aligned} \tag{6}$$

The large vocabulary size that is characteristic of many document corpora creates serious problems of sparsity. A new document is very likely to contain words that did not appear in any of the documents in a training corpus. Maximum likelihood estimates of the multinomial parameters assign zero probability to such words, and thus zero probability to new documents. The standard approach to coping with this problem is to smooth the multinomial parameters, assigning positive probability to all vocabulary items whether or not they are observed in the training set [24]. We apply Laplace smoothing which adds 1 to each feature count results in smoothed update equations:

$$\begin{aligned} \theta_{mk} &= \frac{\sum_v y_{mkv} + 1}{N_m + K}, \\ \eta_{kv} &= \frac{\sum_m y_{mkv} + 1}{\sum_{mv} y_{mkv} + V}, \\ y_{mkv} &= \Psi^{-1}(\log N_m \theta_{mk} \eta_{kv}) - 1. \end{aligned} \tag{7}$$

5.2 Maximum a Posterior Estimation

Another approach to smoothing the multinomial parameters is making the mean of the posterior distribution under a Dirichlet prior on the multinomial parameters, which leads to a maximum a posterior(MAP) probability estimate. We treat θ, η as random variables and add symmetric Dirichlet priors with strengths α for θ and β for η, respectively. It is possible to optimize θ, η for the MAP estimation. The derivation is very similar to the MLE derivation in subsection 5.1 and so is the update equations:

$$\begin{aligned}\hat{\theta}_{mk} &= \frac{\sum_v y_{mkv} + \alpha}{N_m + K\alpha},\\ \hat{\eta}_{kv} &= \frac{\sum_m y_{mkv} + \beta}{\sum_{mv} y_{mkv} + \beta V},\\ \hat{y}_{mkv} &= \Psi^{-1}(\log N_m \theta_{mk} \eta_{kv}) - 1.\end{aligned} \tag{8}$$

Exact inference for this model is computationally complicated because of computing $\psi^{-1}(x)$ in equations 7 and 8 and memory-consuming for recording y_{mkv}. In each iteration, when updating the values for all the variable, computing y_{mkv} involves logarithm functions and inverse function, which are approximated by Taylor's polynomials. Besides, the memory needed to record y_{mkv} is $O(MKV)$. These two aspects lead to great computational difficulties for direct inference.

5.3 Inference with Sampling Distribution

We employ the sampling method to infer the parameters. The idea of sampling is that we do not explicitly solve the parameters directly but infer the values of all the hidden variables first. A sampling distribution is the probability distribution of a given statistic based on a random sample. Sampling distributions provide a major simplification to statistical inference. More specifically, they allow analytical considerations to be based on the sampling distribution of a statistic. Therefore we use sampling distribution to approximate the unknown distribution and learn parameters from the random sample.

Both $\hat{\theta}$ and $\hat{\eta}$ are sufficient statistics of y_{mkv}, so we focus only on sampling them. However, the counts y_{mkv} are integer-valued variables and cannot be sampled directly. Considering the update rules in subsections 5.1 and 5.2, the counts $N_{mk.} = \sum_v y_{mkv}$, $N_{.kv} = \sum_m y_{mkv}$ and $N_{.k.} = \sum_{m,v} y_{mkv}$ should be recorded in each iteration for parameter inference. Let '$z_{mn} = k$' denote assigning topic k to the n^{th} word in document m. All the counts needed to infer parameters depend only on z_{mn}. We may rewrite N_{mkv} as $\sum_{n=1}^{N_m} \delta(z_{mn}, k)\delta(w_{mn}, v)$. Hence, what we will place importance on is sampling topic assignment for each word. Moreover, updating the counts only sweeping through after all samplings in each iteration may affect the rate of convergence [35]. Addressing this problem, in our experiments we update the counts just after an assignment and sample the next topic assignment with the new counts to speed up the rate of convergence. Taking into account the reasons above, we get sampling formulas as follows.

$$\begin{aligned}p(z_{mn} = k) &\propto \frac{N_{mk.} + 1}{N_m + K}\frac{N_{.kv} + 1}{N_{.k.} + V}, \qquad (MLE)\\ p(z_{mn} = k) &\propto \frac{N_{.kv} + \beta}{N_{.k.} + V\beta}\frac{N_{.kv} + \alpha}{N_{.k.} + K\alpha}. \qquad (MAP)\end{aligned} \tag{9}$$

The sampling formulas in Equation 9 are similar to those used for PLSI and LDA just because of the employment of multinomial distribution framework, which means that these algorithms can be made to perform similarly with proper parameter setting [2].

6 EXPERIMENTS

In this section, we empirically evaluate the effectiveness of our model on four datasets, which are representative benchmark in text analysis. Our experiments include topic coherence measurement and document clustering.

Table 2: Statistics of datasets

Dataset	#Documents	#Terms
NIPS	1740	13649
Reuters-21578	8293	18933
TDT2	9394	36771
20NewsHome	18774	61188

6.1 Data sets

Table 2 provides the statistics of the four document corpora.

The Reuters-21578 dataset[1] contains 21578 documents which are grouped into 135 clusters. We use here the ModApte version. Those documents with multiple category labels are discarded. It leaves us with 8293 documents in 65 categories. For ModeApte split, there are 5946 training documents and 2347 testing documents. Compared with TDT2 corpus, the Reuters corpus is more difficult for clustering.

We also use a subset of the original TDT2 corpus. The TDT2 dataset[2]corpus consists of data collected during the first half of 1998 and taken from 6 sources, including 2 newswires (APW, NYT), 2 radio programs (VOA, PRI) and 2 television programs (CNN, ABC). It consists of 11201 on-topic documents which are classified into 96 semantic categories. In TDT2, the content of each cluster is narrowly defined, whereas in Reuters, documents in each cluster have a broader variety of content. Moreover, the Reuters corpus is much more unbalanced, with some large clusters more than 300 times larger than some small ones. In our test, we discard documents with multiple category labels, and only select the categories with more than 10 documents. This left us with 8,213 documents in total.

The NIPS dataset consists of 1740 documents, of which the training set consists of 1557 documents and the test set consists of the remaining 183. The vocabulary contains 12113 words. From the raw data we extracted the words that appear in the vocabulary and discarded stop words from the input.

The 20 Newsgroups dataset[3] is a collection of approximately 20,000 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups. To the best of my knowledge, it was originally collected by Ken Lang, probably for his Newsweeder: Learning to filter netnews paper, though he does not explicitly mention this collection. The 20 newsgroups collection has become a popular data set for experiments in text applications of machine learning techniques, such as text classification and text clustering.

6.2 Experimental Settings

We compare our model with PLSI and LDA on topic quality and document clustering. When implementing topics models, choosing the value K, the number of features, is also an important problem. Different values imply different complexity and maybe lead to heavy computational complexity on large data sets. However, we

[1] http://www.daviddlewis.com/resources/testcollections/reuters21578

[2] http://www.nist.gov/speech/tests/tdt/tdt98/index.html

[3] http://qwone.com/ jason/20Newsgroups/

will not discuss it in this paper but alternatively we fix different values in advance. In our experiments thus far, we fixed the number of topics $K \in \{50, 100, 200, 300\}$. The empirical prior parameters are set for all or part of models. We let $\beta = 0.01$ and $\alpha = 50/K$. We train topic models and estimated the parameters on all the documents.

We run each sampler for 500 iterations with 200 burn-in with the varying values of K, the number of topics. The value of 500 iterations is chosen to guarantee the convergence of posterior distribution so that topics are nearly drawn from the true distribution.

6.3 Evaluating topics

In this section, we study the quality of topics from different models. Topic models give no guarantees on well interpretable output, which extract topics from word counts in documents without requiring any semantic annotations. Therefore, coherence measures [36, 37, 39] were proposed and have been approved to distinguish between good and bad topics based on top words with respect to interpretability.

6.3.1 Coherence measures. Coherence measures compute a sum of scores over pairs of words from top words of a given topic:

$$coherence = \sum_{i<j} score(w_i, w_j).$$

The state-of-the-art measures in terms of topic coherence are the intrinsic measure UMass and the extrinsic measure UCI, both based on the same high-level idea.

The UCI-coherence measures the coherence of a topic based on pointwise mutual information(PMI) using large scale text data sets from external sources. Given the T most probable words of a topic k, $(w_1, \ldots, w_T)$, PMI-Score measures the pairwise association between them.

$$PMI(w_i, w_j) = \log \frac{P(w_i, w_j) + \frac{1}{M}}{P(w_i)P(w_j)}$$

where $P(w_i, w_j)$, $P(w_i)$ and $P(w_j)$ are the probabilities of co-occurring words pair (w_i, w_j) and w_i is estimated empirically from the external data sets, respectively. The smoothing count $\frac{1}{M}$ is added to avoid calculating the logarithm of zero. The UCI-coherence is calculated by:

$$UCI - coherence(topic) = \sum_{i=2}^{T} \sum_{j=1}^{i-1} PMI(w_i, w_j). \quad (10)$$

The coherence based on pointwise mutual information (PMI) gave large correlations with human ratings. The measure is extrinsic as it uses empirical probabilities from an external corpus such as Wikipedia. In our experiments, we extract topics and then compute UCI-coherence on the same dataset. Since these external data sets are model-independent, UCI-Score is fair for all the topic models.

The UMass coherence measure([34, 39]) takes the set of T (here T is set to 10) top words of a topic and sum a confirmation measure over all word pairs. Given an ordered list of words $(w_1, ..., w_T)$, the UMass-coherence is calculated by:

$$UMass - coherence(topic) = \sum_{i=2}^{T} \sum_{j=1}^{i-1} \log \frac{p(w_i, w_j) + \frac{1}{M}}{p(w_j)}, \quad (11)$$

Table 3: Average topic coherence results on the NIPS dataset

		100	200	300
PLSI	UCI	0.3284	0.3332	0.3018
	UMass	-1.082	**-1.030**	**-1.1065**
LDA	UCI	**0.3719**	0.4120	**0.4522**
	UMass	-1.1020	-1.2474	-1.4609
PDM-MAP	UCI	0.3618	**0.4192**	0.4392
	UMass	**-0.7252**	-1.3581	-1.3692

where $P(w_i, w_j)$ and $P(w_i)$ are the probabilities of co-occurring words pair (w_i, w_j) and w_i is estimated based on document frequencies of the original documents used for learning the topics. A boolean document model is assumed to estimate word probabilities p, i.e., $p(w_i, w_j)$ is the ratio of number of documents containing both words (w_i, w_j) and the total number of documents in the corpus. The smoothing count $\frac{1}{M}$ is added to avoid calculating the logarithm of zero.

The score function is not symmetric as it is an increasing function of the empirical probability $p(w_j|w_i)$, where w_i is more common than w_j, words being ordered by decreasing frequency $p(w|k)$. So this score measures how much, within the words used to describe a topic, a common word is in average a good predictor for a less common word.

For the two topic measures, the higher coherence score corresponds to a better topic quality.

6.3.2 Topics Evaluations. Table 3 shows the average coherence measures of topic models with different fixed topic numbers on the NIPS dataset. It clearly shows the difference between the quality of the topics extracted by different models. For UCI-coherence our model always performs better than PLSI and closely to LDA while for UMass-coherence PLSI performs better than LDA and our model. Besides, the three models get better results as the number of topics get larger. In general, our proposed model has similar performance with LDA.

Table 4 and Table 5 list 10 most coherent topics sorted by UCI-coherence or by UMass-coherence, respectively. These tables clearly show the topic ranking and topic representation. In the list sorted by UCI-coherence, 6 of 10 topics were extracted by PDM-MAP and 4 of 10 by LDA and 0 of 10 by PLSI. This result can also be easily tested by human judgement. For example, words in the the first topic { learning, search, code, number, algorithm, competitive, coding, genetic} have strong coherence between them and so that the topic has a higher UCI-coherence score.

It is interesting that PLSI has a much better performance on UMass-coherence evaluation than PDM-MAP and LDA. Of the 10 high-score coherent topics by UMass-coherence, 6 topics were extracted by PLSI, 3 topics by PDM-MAP and 1 topic by LDA. It seems that the topics with high coherence scores by PLSI have apparent coherence. For example, in the topic {propagation, back, net, training, network, neural, learning, performance}, 'network' has apparent coherence with the other words of the same topic. This denies the common statement that PLSI cannot extract better topics than LDA. What results in this appearance should be deeply discovered in the future.

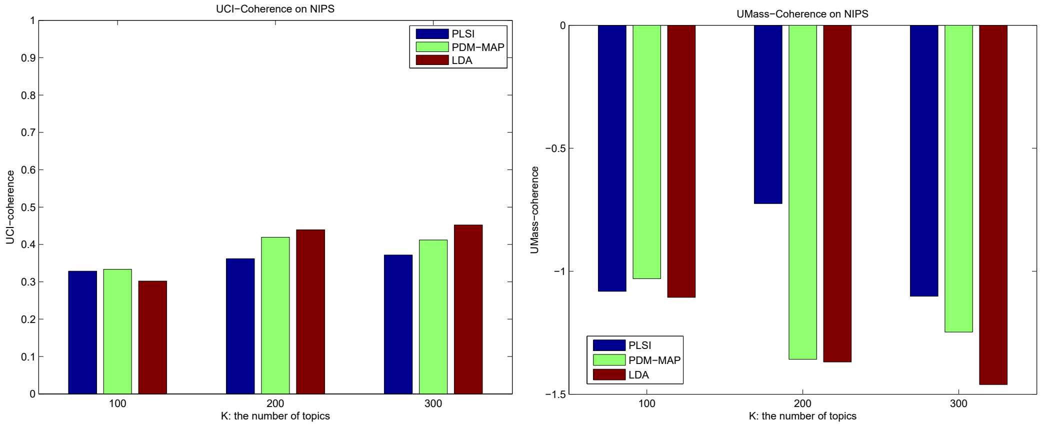

Figure 2: Average topic coherence results for LDA, PLSI and PDM on the NIPS dataset with K(the number of topics) ∈ $\{100, 200, 300\}$.

Table 4: Topics ranking sorted by UCI-coherence. We list 10 topics of 100 topics which have the largest UCI-coherence and each row gives a topic represented by the top ten words. The first column indicates which model created the topic.

model								
PDM	learning	search	code	number	algorithm	competitive	coding	genetic
PDM	time	neurons	model	neuron	frequency	spike	firing	synaptic
LDA	rate	figure	adaptation	time	constant	shown	change	rates
PDM	image	images	object	recognition	features	feature	objects	set
LDA	matrix	linear	vector	nonlinear	space	diagonal	matrices	operator
LDA	threshold	neural	binary	boolean	size	depth	gate	number
PDM	model	data	models	distribution	gaussian	parameters	probability	mixture
LDA	optimal	error	complexity	criterion	number	parameters	pruning	term
PDM	horizontal	top	vertical	bottom	location	moving	moving	center
PDM	cells	field	visual	model	cell	cortex	orientation	receptive

Table 5: Topics ranking: 10 most coherent topics by UMass-coherence. We rank 10 topics of 100 topics Each row gives a topic represented by the eight words.

model								
PLSI	propagation	back	net	training	network	neural	learning	performance
PLSI	network	networks	neural	problem	function	number	results	set
PLSI	nodes	node	network	input	inputs	networks	output	neural
PLSI	sequence	sequences	recurrent	time	training	networks	neural	test
PDM	training	error	set	learning	data	prediction	examples	test
PDM	simulation	occurs	simulated	information	increases	behavior	initially	time
PDM	learning	state	rules	rule	sequence	recurrent	input	sequences
LDA	algorithm	algorithms	convergence	step	update	learning	iteration	results
PLSI	rules	rule	learning	input	set	knowledge	neural	output
PLSI	network	units	hidden	input	unit	layer	output	training

6.4 Document Clustering

To measure the quality of the learned topical representations from different models, we use k-means document clustering problem to see how accurate and discriminative the features obtained by different models are. k-means clustering aims to partition N observations into k clusters in which each observation belongs to the cluster with the nearest mean.

6.4.1 Experiment Description. Document clustering is a well researched area that has several traditional methods available. In the text clustering problem, we wish to classify a document into two or more mutually exclusive classes. A challenging aspect of the document classification problem is feature selection and it is essential in the document clustering problem. Treating all words as features yields a rich but very large feature set, which often causes great complexity. One way to reduce this feature set is to use topic models for dimensionality reduction and this is our focus in this section. PLSI [26] and LDA [4] have also been reported to provide improved performance in almost all cases. For each document, its topical representation is a vector. We use PLSI, LDA and PDM to reduce this feature set, which can all reduce any document to a fixed set of features. The difference is that PDM reduces it by the multidimensional Poisson parameters associated with the document while PLSI and LDA by the multidimensional distribution associated with the document. In these experiments, we split the data set into training and test subsets, and employed the k-means algorithm on the low-dimensional representations provided by PLSI, LDA and PDM, respectively. It is of interest to see how much discriminatory information we leave in reducing the document description to topic-based features.

6.4.2 Evaluation Metric. The clustering result is evaluated by comparing the obtained label of each sample with that provided by the data set. The accuracy(AC) and the normalized mutual information metric(NMI) are used to measure the clustering performance[9].

Given a document d, let c_m and r_m be the obtained cluster label and the label by the corpus, respectively. The AC is defined as

$$AC = \frac{\sum_{m=1}^{M} \delta(c_m, r_m)}{M},$$

where M is the total number of documents and and $\delta(x, y)$ is the delta function that equals 1 if $x = y$ and equals 0 otherwise. Let X denote the set of clusters obtained from the ground truth and Y obtained from our models. $H(X)$, $p(x)$ and $p(x, y)$ denote the entropy of X, the probabilities that a document arbitrarily selected from the corpus belongs to the clusters x, and the joint probability that the arbitrarily selected document belongs to the clusters x as well as y at the same time, respectively. The mutual information metric $MI(X, Y)$ and the normalized mutual information metric $NMI(X, Y)$ are defined as follows:

$$MI(X, Y) = \sum_{x \in X, y \in Y} p(x, y) \log_2 \frac{p(x, y)}{p(x)p(y)},$$

$$NMI(X, Y) = \frac{MI(X, Y)}{\max\{H(X), H(Y)\}}.$$

The value of NMI ranges from 0 to 1 and a larger value means stronger independence.

6.4.3 Performance Evaluations. Figure 3 and Table 6 show document clustering performance using the TDT2, Reuters21578 and 20NewsHome datasets, respectively. The evaluations were conducted with the topics(features) numbers varying in $\{50, 100, 200, 300\}$. The similar performance between PDM and LDA confirms PDM's validity and effectiveness.

We can see that the clustering accuracy is in line with the NMI for the three datasets. All the models achieve good accuracy on the datasets of strong independence and achieve poor accuracy on the datasets of weak independence, especially on the Reuters21578 dataset, which supports the common statement that the Reuters21578 dataset is more difficult to cluster.

Regardless of the data sets and the numbers of topics, clustering with based on PDM features[4] always has better performance than PLSI features and has similar performance with LDA features, which accords to that PDM and LDA employ multinomial distribution framework and sampling formulas. Specially, clustering using PDM features has a performance improvement on the Reuters21578 dataset than on other datasets, which was thought more difficult to cluster. There is also an interesting point that clustering using PDM-MLE features has a performance improvement on the 20NewsHome dataset. One possible reason is Laplace smoothing method which assigns non-zero probability to new documents.

The clustering results demonstrate that the topic-based representation provided by PDM can be thought as a filtering algorithm for feature selection in text analysis.

7 CONCLUSIONS

We have described an intuitive and easy-to-understand topic model based on Poisson decomposition. The model is a statistic model that builds on Raikov's theorem and aims to obtain their numerical characters of documents. We can also view PDM as a dimensionality reduction technique with proper underlying generative probabilistic semantics, in the spirit of LSI and LDA. As a simple statistics model, PDM has similar performance with LDA and better than PLSI, which were shown in the experimental results. We should note that our approach as well as other existing topic models focuses on high frequent terms which leads to cover difficultly the long-tail semantic word sets. However, our approach was developed in the frame of matrix factorization and can be easily extended. We expect that the coherence improvements achieved by extensibility of PDM will provide long tail topics from a small amount of topics. Finally, developing more coherent topic models as well as evaluating topic coherence is still a tough task and breaking the probabilistic frame may be a good attempt.

8 ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China(Grant No. 61672161, 61272480), and in part by the grants of Australian Research Council Discovery Project (DP170104747).

REFERENCES

[1] David J Aldous. 1985. *Exchangeability and related topics.* Springer.
[2] Arthur Asuncion, Max Welling, Padhraic Smyth, and Yee Whye Teh. 2009. On smoothing and inference for topic models. (2009), 27–34.
[3] David M Blei. 2012. Probabilistic topic models. *Communications of The ACM* 55, 4 (2012), 77–84.
[4] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *the Journal of machine Learning research* 3 (2003), 993–1022.
[5] Jordan Boydgraber, David M Blei, and Xiaojin Zhu. 2007. A Topic Model for Word Sense Disambiguation. (2007).
[6] S R K Branavan, Harr Chen, Jacob Eisenstein, and Regina Barzilay. 2009. Learning document-level semantic properties from free-text annotations. *Journal of Artificial Intelligence Research* 34, 1 (2009), 569–603.
[7] Thorsten Brants, Francine R Chen, and Ioannis Tsochantaridis. 2002. Topic-based document segmentation with probabilistic latent semantic analysis. (2002), 211–218.
[8] Wray Buntine. 2002. Variational Extensions to EM and Multinomial PCA. *Lecture Notes in Computer Science* (2002), 23–34.

[4] For convenience, we denote PDM features as using the features of datasets based on the PDM-MLE and PDM-MAP models.

Table 6: Clustering performance on three datasets. Each entry is the clustering accuracy(left) and NMI(right) of the column method on the corresponding row topic numbers .

a Clustering performance on Reuters21578

	PLSI	PDM-MLE	PDM-MAP	LDA
K=50	(.4645, .4685)	(**.5502**, .5509)	(.5224, .5327)	(.5324, .5425)
K=100	(.4571, .4515)	(**.5283**, .5366)	(.5021, .5380)	(.5010, .5247)
K=200	(.4424, .4016)	(.4623, .4719)	(**.4961**, .4967)	(.4676, .4770)
K=300	(.3436, .3239)	(.4973, .4943)	(.4840, .4771)	(**.5014**, .4947)

b Clustering performance on TDT2

	PLSI	PDM-MLE	PDM-MAP	LDA
K=50	(.5194, .6338)	(.5590, .6757)	(**.5964**, .6875)	(.5877, .6703)
K=100	(.4766, .6308)	(.6792, .7368)	(**.6854**, .7295)	(.6050, .7101)
K=200	(.5046, .6317)	(.5770, .7082)	(**.6021**, .7127)	(.5994, .6988)
K=300	(.5277, .6370)	(.6231, .7242)	(.6436, .6870)	(**.6604**, .7380)

c Clustering performance on 20NewsHome

	PLSI	PDM-MLE	PDM-MAP	LDA
K=50	(.4645, .4685)	(**.5502**, .5509)	(.5224, .5327)	(.5324, .5425)
K=100	(.4571, .4515)	(**.5283**, .5366)	(.5021, .5380)	(.5010, .5247)
K=200	(.4424, .4016)	(.4623, .4719)	(**.4961**, .4967)	(.4676, .4770)
K=300	(.3436, .3239)	(.4973, .4943)	(.4840, .4771)	(**.5014**, .4947)

[9] Deng Cai, Xiaofei He, Senior Member, and Jiawei Han. 2009. Locally consistent concept factorization for document clustering. *IEEE Transactions on Knowledge and Data Engineering* (2009), 902–913.

[10] John Canny. 2004. GaP: a factor model for discrete data. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 122–129.

[11] Laurent Charlin, Rajesh Ranganath, James McInerney, and David M Blei. 2015. Dynamic Poisson Factorization. In *Proceedings of the 9th ACM Conference on Recommender Systems.* ACM, 155–162.

[12] Chaitanya Chemudugunta, Padhraic Smyth, and Mark Steyvers. 2006. Modeling general and specific aspects of documents with a probabilistic topic model. (2006), 241–248.

[13] Wenyen Chen, Dong Zhang, and Edward Y Chang. 2008. Combinational collaborative filtering for personalized community recommendation. (2008), 115–123.

[14] Abhinandan S Das, Mayur Datar, Ashutosh Garg, and Shyam Rajaram. 2007. Google news personalization: scalable online collaborative filtering. *international world wide web conferences* (2007), 271–280.

[15] Bruno De Finetti. 1974. *Theory of Probability: A Critical Introductory Treatment. Transl. by Antonio Machi and Adrian Smith.* J. Wiley.

[16] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science* 41, 6 (1990), 391.

[17] Prem Gopalan, Jake M Hofman, and David M Blei. 2015. Scalable Recommendation with Hierarchical Poisson Factorization. (2015).

[18] Prem Gopalan, Francisco JR Ruiz, Rajesh Ranganath, and David M Blei. 2014. Bayesian nonparametric Poisson factorization for recommendation systems. *Artificial Intelligence and Statistics (AISTATS)* 33 (2014), 275–283.

[19] Prem K Gopalan, Laurent Charlin, and David Blei. 2014. Content-based recommendations with Poisson factorization. In *Advances in Neural Information Processing Systems.* 3176–3184.

[20] Thomas L Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences* 101, suppl 1 (2004), 5228–5235.

[21] Thomas L Griffiths, Mark Steyvers, David M Blei, and Joshua B Tenenbaum. 2004. Integrating topics and syntax. (2004), 537–544.

[22] Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 50–57.

[23] Thomas Hofmann. 2004. Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems* 22, 1 (2004), 89–115.

[24] Frederick Jelinek. 1997. *Statistical Methods for Speech Recognition.* MIT Press.

[25] Jing Jiang and Chengxiang Zhai. 2006. Extraction of coherent relevant passages using hidden Markov models. *ACM Transactions on Information Systems* 24, 3 (2006), 295–319.

[26] Thorsten Joachims. 1998. *Text categorization with support vector machines: Learning with many relevant features.* Springer.

[27] Ralf Krestel, Peter Fankhauser, and Wolfgang Nejdl. 2009. Latent dirichlet allocation for tag recommendation. (2009), 61–68.

[28] Erich L Lehmann and Joseph P Romano. 2006. *Testing statistical hypotheses.* Springer Science & Business Media.

[29] Chenghua Lin and Yulan He. 2009. Joint sentiment/topic model for sentiment analysis. (2009), 375–384.

[30] Yang Liu, Xiangji Huang, Aijun An, and Xiaohui Yu. 2007. ARSA: a sentiment-aware model for predicting sales performance using blogs. (2007), 607–614.

[31] Yue Lu and Chengxiang Zhai. 2008. Opinion integration through semi-supervised topic modeling. *international world wide web conferences* (2008), 121–130.

[32] Eugene Lukacs. 1970. Characteristics functions. *Griffin, London* (1970).

[33] Qiaozhu Mei, Xu Ling, Matthew Wondra, Hang Su, and Chengxiang Zhai. 2007. Topic sentiment mixture: modeling facets and opinions in weblogs. *international world wide web conferences* (2007), 171–180.

[34] David Mimno, Hanna M Wallach, Edmund Talley, Miriam Leenders, and Andrew Mccallum. 2011. Optimizing semantic coherence in topic models. (2011), 262–272.

[35] Radford M. Neal and Geoffrey E. Hinton. 1998. *A View of the Em Algorithm that Justifies Incremental, Sparse, and other Variants.* Springer Netherlands, Dordrecht, 355–368. https://doi.org/10.1007/978-94-011-5014-9_12

[36] David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. 2010. Automatic evaluation of topic coherence. (2010), 100–108.

[37] Derek Ocallaghan, Derek Greene, Joe Carthy, and Padraig Cunningham. 2015. An analysis of the coherence of descriptors in topic modeling. *Expert Systems With Applications* 42, 13 (2015), 5645–5657.

[38] D Raikov. 1938. On the decomposition of Gauss and Poisson laws. *Izv. Akad. Nauk SSSR Ser. Mat.* 1 (1938), 91–124.

[39] Michael Roder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the Space of Topic Coherence Measures. (2015), 399–408.

[40] L I Rongmei, Rianne Kaptein, Djoerd Hiemstra, and Jaap Kamps. 2008. Exploring Topic-based Language Models for Effective Web Information Retrieval. *IEEE Electron Device Letters* (2008), 65–71.

[41] Ivan Titov and Ryan Mcdonald. 2008. Modeling online reviews with multi-grain topic models. *international world wide web conferences* (2008), 111–120.

[42] Kristina Toutanova and Mark Johnson. 2008. A Bayesian LDA-based Model for Semi-Supervised Part-of-speech Tagging. (2008), 1521–1528.

[43] Hanna M Wallach. 2006. Topic modeling: beyond bag-of-words. (2006), 977–984.

[44] Xuerui Wang, Andrew Mccallum, and Xing Wei. 2007. Topical N-Grams: Phrase and Topic Discovery, with an Application to Information Retrieval. (2007), 697–702.

[45] Xing Wei and W Bruce Croft. 2006. 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.* ACM, 178–185.

[46] Ding Zhou, Jiang Bian, Shuyi Zheng, Hongyuan Zha, and C Lee Giles. 2008. Exploring social annotations for information retrieval. *international world wide web conferences* (2008), 715–724.

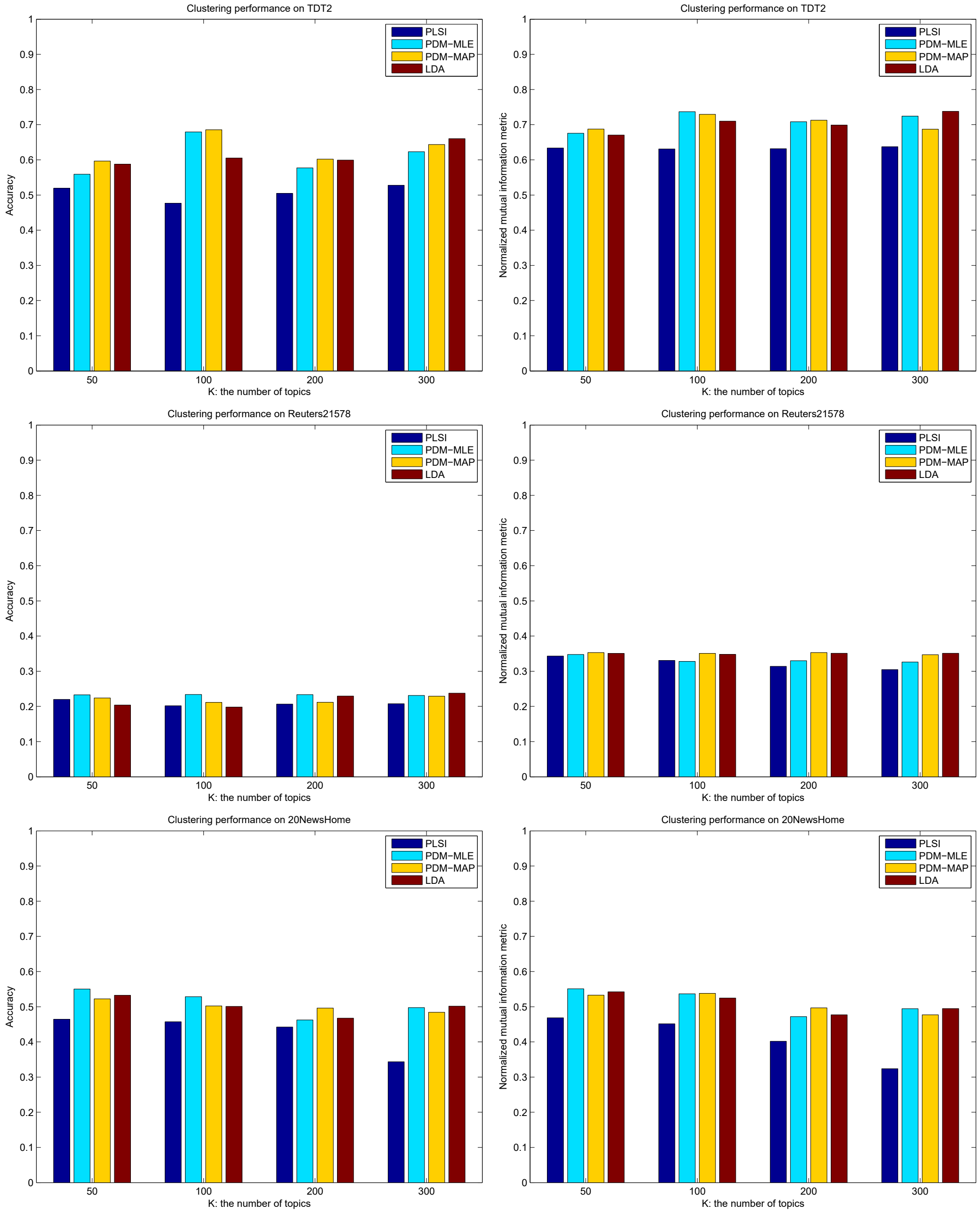

Figure 3: Accuracy (left) and normalized mutual information (right) results on the TDT2(top), Reuters21578(middle) and 20NewsHome(bottom) data sets for PDM, LDA and PLSI.