genes

MDPI

Article

A Multi-Label Supervised Topic Model Conditioned on Arbitrary Features for Gene Function Prediction

Lin Liu [1], Lin Tang [2,*], Xin Jin [3] and Wei Zhou [3,*]

[1] School of Information, Yunnan Normal University, Kunming 650500, China; liulinrachel@163.com
[2] Key Laboratory of Educational Informatization for Nationalities Ministry of Education, Yunnan Normal University, Kunming 650500, China
[3] School of Software, Yunnan University, Kunming 650091, China; xinxin_jin@163.com
[*] Correspondence: maitanweng2@163.com (L.T.); zwei@ynu.edu.cn (W.Z.); Tel.: +86-087-165-739-989 (L.T.)

Received: 30 November 2018; Accepted: 10 January 2019; Published: 17 January 2019

Abstract: With the continuous accumulation of biological data, more and more machine learning algorithms have been introduced into the field of gene function prediction, which has great significance in decoding the secret of life. Recently, a multi-label supervised topic model named labeled latent Dirichlet allocation (LLDA) has been applied to gene function prediction, and obtained more accurate and explainable predictions than conventional methods. Nonetheless, the LLDA model is only able to construct a bag of amino acid words as a classification feature, and does not support any other features, such as hydrophobicity, which has a profound impact on gene function. To achieve more accurate probabilistic modeling of gene function, we propose a multi-label supervised topic model conditioned on arbitrary features, named Dirichlet multinomial regression LLDA (DMR-LLDA), for introducing multiple types of features into the process of topic modeling. Based on DMR framework, DMR-LLDA applies an exponential a priori construction, previously with weighted features, on the hyper-parameters of gene-topic distribution, so as to reflect the effects of extra features on function probability distribution. In the five-fold cross validation experiment of a yeast datasets, DMR-LLDA outperforms the compared model significantly. All of these experiments demonstrate the effectiveness and potential value of DMR-LLDA for predicting gene function.

Keywords: multi-label classification; topic model; gene function; probability distribution; Dirichlet-multinomial Regression

1. Introduction

As the main component of a cell, proteins are the most essential and versatile material of life. Thus, the research on protein functions is of great importance for the development of new drugs, better crops, and the development of synthetic biochemical [1]. In recent years, new protein function prediction methods using machine learning algorithms have proliferated, based on various known information about proteins, and have increasingly become important long-standing research works in the post-genomic era. From the point of molecular biology, a protein is the product of a gene after the process of transcribing, translating, and post-translational modifying. Even though the real function of a gene is to encode one or more proteins executing practical functions, the function of a gene product has usually been regarded as the native function of the gene in gene-level experiments. Therefore, we do not distinguish between gene function and protein function in this paper, which are known collectively as gene function.

The most common computational approach for gene function prediction is to transfer the gene function into some specific features from their sequence or structure similarity, such as BLAST [2]. In addition to sequence similarity, many gene function prediction methods have been exploited in recent years as the additional information extracted from proteins, such as protein structure [3], protein motif, biophysical properties [4], and integrated heterogeneous data sources [5]. In reference [3], Evangelia et al. extract novel shape features from protein structures in the form of local (per amino acid) distribution of angles and amino acid distances, respectively. Each of the multi-channel feature maps is introduced into a deep convolutional neural network (CNN) for function prediction, and the outputs are fused through support vector machines or a correlation-based *k*-nearest neighbor classifier. In addition, automatic prediction using protein–protein similarity information can be further supplemented by experimental data [6,7]; this kind of method assumes that the closely related proteins (or genes) share similar functional annotations on the basis of network structure information. Researchers have made the relevant literature reviews of computational methods on gene function prediction in references [8–10].

From the point of machine learning algorithms, predicting gene function based on various data sources is a problem of classification in nature. A gene can be viewed as an instance to be classified—various kinds of data sources (such as an amino acid sequence, textual repositories, and motifs) can be organized into a feature space, so that each gene is represented as a set of attribute values; a function (such as a gene ontology (GO) term [11]) is regarded as a label. As a gene is always annotated by several functions, gene function prediction is actually a process of multi-label classification: a multi-label classifier is trained firstly on constructed attribute features and annotated genes, and then is used to predict function annotations for unannotated genes. From the above analysis, we believe that many multi-label classification algorithms have great potential to predict gene function, such as a support vector machine (SVM), neural network, and decision tree. In reference [12], Celine Vens et al. proposed three multi-label classifiers based on a hierarchical decision tree, and the experimental results from 24 datasets show that these classifiers are powerful and effective for gene function prediction.

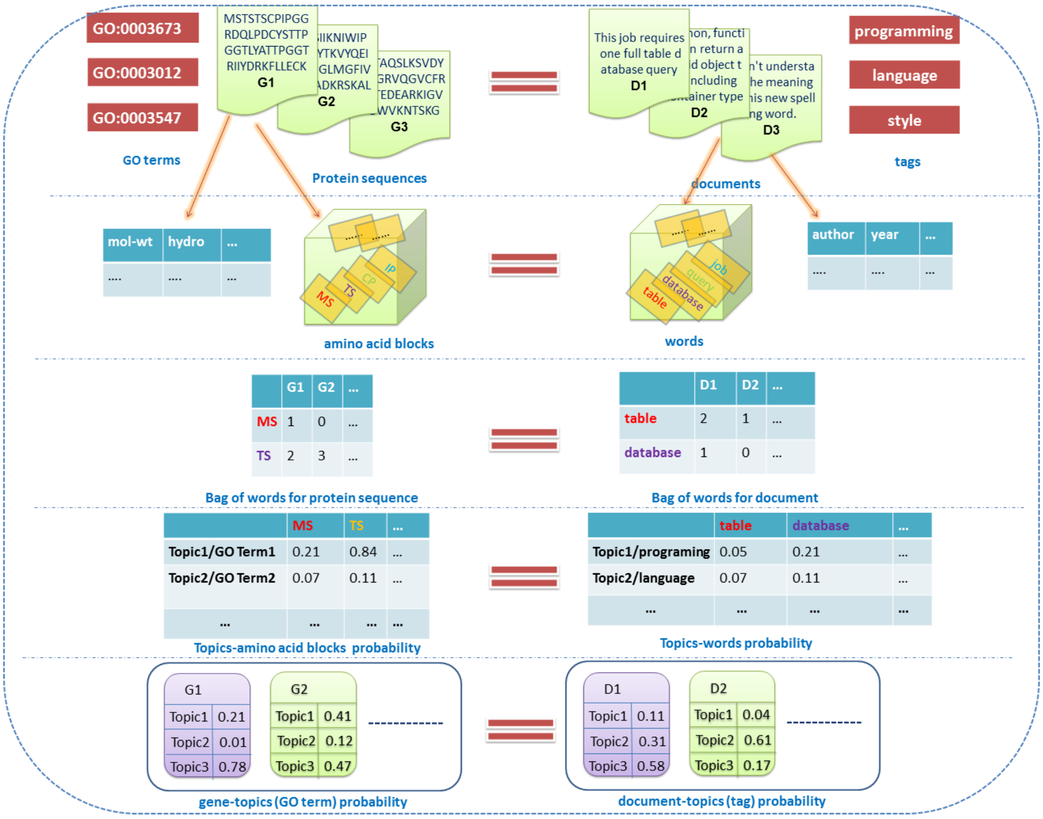
In addition to traditional machine learning algorithms, a topic model is a kind of probabilistic generative model that has been applied into gene function prediction. In reference [13], Liu et al. introduced a typical multi-label supervised topic model into gene function prediction, which was called labeled latent Dirichlet allocation (LLDA) and is proposed in reference [14] for text mining. This research is the first effort to apply a multi-label supervised topic model into gene function prediction. Compared with traditional multi-label classification models, LLDA can model a function label as a topic, and thus can not only work out the function probability distributions over gene instances effectively, but can also directly provide the word probability distributions over functions. Nonetheless, the direct application of LLDA on a gene function dataset can only utilize protein sequence data by formalizing the sequences into a bag of words (BoW), and then the constructed bag of words is used for topic modeling. In other words, due to the restrictions of BoW construction in topic modeling, the feature space was constructed on sequence data rather than multiple biological data. However, we can see from the above paragraph that there are various protein features, such as hydrophobicity and the polarity of amino acids, which have a profound impact on gene structure and function. Apparently, the introduction of multiple kinds of gene features in a multi-label supervised topic model can improve the accuracy of gene function prediction.

Inspired by the application of a multi-label topic model in gene function prediction and a topic model conditioned on arbitrary features named the Dirichlet multinomial regression latent Dirichlet allocation (DMR-LDA) [15], we propose a DMR-LLDA model, which introduces a DMR framework into an LLDA model. Firstly, we describe DMR-LLDA for gene function prediction problem formulation. Then the generative process and the inference algorithm of DMR-LLDA are described. This model is fully compatible with both discrete and continuous features, whose inference is relatively simple. In a five-fold cross validation experiment on verified gene function prediction, DMR-LLDA significantly outperformed LLDA. In addition, the impact of feature variables on prior parameters and the comparison between two kinds of inference algorithms are shown in experimental data. All these experimental results demonstrate the effectiveness and potential value of DMR-LLDA for predicting gene function.

2. Methods

2.1. Related Definitions and Notations

In this paper, the topic modeling method of gene function prediction reported by reference [13] is utilized. We consider each gene to be a document [16], and GO terms (topics) are shared by a document collection. Meanwhile, we view the extra gene features, except for the bag of amino acid words, as the metadata, such as authors and dates of documents. Therefore, the introduction of extra gene features into topic modeling is similar to introducing metadata into the topic modeling of documents, and the type of metadata may be discrete or continuous. To better understand the practical application of our method, the relationship of text topic modeling and gene function predicting is illustrated by Figure 1.


GO:0003673
GO:0003012
GO:0003547
GO terms
MSTSTSCPIPGG RDQLPDCYSTTP GGTLYATTPGGT RIIYDRKFLLECK
G1
IIKNIWIP YTKVYQEI GLMGFIV ADKRSKAL
G2
FAQSLKSVDY GRVQGVCFR EDEARKIGV WVKNTSKG
G3
Protein sequences
This job requires one full table d atabase query
D1
non, functi n return a d object t ncluding ...ainer type
D2
n't understa he meaning his new spell ng word.
D3
documents
programming
language
style
tags
mol-wt
hydro
...
....
amino acid blocks
IP
CP
TS
MS
words
job
query
database
table
author
year
Bag of words for protein sequence
Bag of words for document
Topics-amino acid blocks probability
Topics-words probability
gene-topics (GO term) probability
document-topics (tag) probability

mol-wt	hydro	...
....	....	...

author	year	...
....	....	...

	G1	G2	...
MS	1	0	...
TS	2	3	...

	D1	D2	...
table	2	1	...
database	1	0	...

	MS	TS	...
Topic1/GO Term1	0.21	0.84	...
Topic2/GO Term2	0.07	0.11	...
...	...	...	...

	table	database	...
Topic1/programing	0.05	0.21	...
Topic2/language	0.07	0.11	...
...	...	...	...

G1	
Topic1	0.21
Topic2	0.01
Topic3	0.78

G2	
Topic1	0.41
Topic2	0.12
Topic3	0.47

D1	
Topic1	0.11
Topic2	0.31
Topic3	0.58

D2	
Topic1	0.04
Topic2	0.61
Topic3	0.17

Figure 1. The relationship between protein function prediction and text topic modeling. IP, CP, TS, MS and so on, represent ‘words’, each of which is composed by two amino acid alphabets. Each GO term is started by ‘GO:’.

In Figure 1, the right part describes the topic modeling concept of text data, and the left part describes the related concept of gene function data. For all topic models, there are three key concepts: “documents”, “words”, and “topics”. In addition, the supervised topic model introduces “labels” for each document, and the proposed DMR-LLDA model introduces “features” for each document. Therefore, these concepts can now be reformulated with more detail, as follows.

2.1.1. Documents

For text data (right part of Figure 1), document collection is composed of several documents numbered D1 to Dn. In the other side (left part of Figure 1), the gene dataset is composed of several protein sequences, numbered G1 to Gn. Therefore, a document is equivalent to a gene in our model. We suppose that there are D genes in a gene set, which compose the gene space $\mathbb{D} = \{1, \ldots, D\}$, and the gene sample set $\mathcal{X}$ including D genes can be represented as $\mathcal{X} = \{\mathcal{X}_d\}_{d=1}^{D}$, and $\mathcal{X}_d$ denotes a gene sample.

2.1.2. Labels

For text data (right part of Figure 1), each document is labeled by one or more tags, such as "programming" and "language". On the other side (left part of Figure 1), each gene is annotated by several GO terms, such as "GO:0003012" and "GO:0003547". Therefore, a document tag is equivalent to a GO term in our model, and all of them are called "labels". In this paper, the gene function label space is expressed as $\mathbb{L} = \{1, \ldots, L\}$. Meanwhile, the observed labels of each gene are described by a sparse binary vector $\mathbf{\Lambda}_d = \{\Lambda_{dl}\}_{l=1}^{L}$, which is defined as follows:

$$\Lambda_{dl} = \begin{cases} 1, l \in \mathbb{L}_d \\ 0, l \notin \mathbb{L}_d \end{cases} \tag{1}$$

where, $\mathbb{L}_d$ represents the label sub-space of gene $\mathcal{X}_d$: $\mathbb{L}_d \subseteq \mathbb{L}$.

2.1.3. Words

For text data (right part of Figure 1), word terms are the main component of a document, such as the words "table" and "database". On the other side (left part of Figure 1), we consider a protein sequence to be a text string, which is defined by a fixed 20 amino acid alphabet (G,A,V,L,I,F,P, Y,S,C,M,N,Q,T,D,E,K,R,H,W). Correspondingly, amino acid blocks are the main components of a protein sequence, which is composed by two or more amino acid alphabets, such as "MS" and "TS". Therefore, a word term is equivalent to an amino acid block in our model, and all of them are called "words". Meanwhile, all of the words constitute a vocabulary. In this paper, the amino acid words space is represented as $\mathbb{W} = \{1, \ldots, W\}$. For a gene $\mathcal{X}_d$, $\mathcal{X}_d = \{\mathbf{x}_{dn}\}_{n=1}^{N_d}$ denotes that the dth gene is composed by N_d observed word samples, and $\mathbf{x}_{dn}$ is one of word samples.

2.1.4. Topics

For text data (right part of Figure 1) and gene function data (left part of Figure 1), a "topic" is viewed as a probability distribution over a fixed vocabulary. Taking the text data as an example, the probabilities of the word "table" over "topic 1" are 0.05. For the gene function data, the probabilities of amino acid block MS over "topic 1" are 0.21. Obviously, topics are latent and needed to be inferred by topic modeling. In this paper, the global topic space includes T topics, which is represented as $\mathbb{T} = \{1, \ldots, T\}$. According to the definition of an LLDA model, there is a one-to-one correspondence between label and topic—therefore, $\mathbb{L} \triangleq \mathbb{T}$ ($\triangleq$ represents equivalent relationship between two space), $T = |\mathbb{T}| = L = |\mathbb{L}|$.

2.1.5. Features

For text data (right part of Figure 1), the metadata of a document can be viewed as document features, such as the tags "author" and "publish year of document". On the other side (left part of Figure 1), each gene has several extra features, except for its sequence string, such as molecular weight and hydrophobicity. Therefore, the metadata of a document tag is equivalent to an extra feature of the gene in our model, and all of them are called "features". In this paper, the feature space composed by gene features is expressed as $\mathbb{F} = \{1, \ldots, F\}$. Therefore, there is a set of observed features for gene $\mathcal{X}_d$, which can be represented as a feature vector: $\mathbf{y}_d = \left\{y_{df}\right\}_{f=1}^{F}$.

2.1.6. Others

In addition to the above five concepts, there are three other concepts illustrated in Figure 1. Firstly, the BoW, which is a word–document matrix and the input of the topic model. In an instance in the right part of Figure 1, the word "table" appears two times in document D1. Likewise, the word "MS" appears one time in gene G1. In other words, the element of the BoW represents the times of each word in each document. Meanwhile, there are two probability matrices that appear in Figure 1: one is the topic (label)–word probability matrix, and the other is the document (gene)–topic probability matrix. All of them are represented as parameter vectors for each topic or gene in the topic model.

A topic corresponds to a multinomial distribution of word space $\mathbb{W}$, whose parameter vector is $\boldsymbol{\theta}_t = \{\theta_{tw}\}_{w=1}^{W}$, and θ_{tw} is the probability of word w under topic t; a gene $\mathcal{X}_d$ corresponds to a multinomial distribution of the topics space $\mathbb{T}$, whose parameter vector is $\boldsymbol{\pi}_d = \{\pi_{dt}\}_{t=1}^{T}$, and π_{dt} is the topic weight of topic t under gene $\mathcal{X}_d$. Finally, we utilize a feature parameter vector $\boldsymbol{\beta}_t = \left\{\beta_{tf}\right\}_{f=1}^{F}$ to represent the relationship between features (f) and topics (t) in making features influence the choice of topic.

Note that the shared parameters of a whole gene set, such as topic–word parameter $\boldsymbol{\theta}$, are called "global parameters" in this paper. Correspondingly, the parameter of one gene is called a local parameter, such as gene–topic (label) parameter $\boldsymbol{\pi}$.

2.2. Overview of the Dirichlet–Multinomial Regression Latent Dirichlet Allocation Topic Modeling Process

Based on the above notation, we can provide the description of a gene function dataset as follows.

The gene $\mathcal{X}_d$ is composed of N_d, which are observed samples, and the word index of each sample x_{dn} comes from the vocabulary $\mathbf{w}_{dn} \in \mathbb{W}_d$. Thus, the gene can also be represented as $\mathcal{W}_d = \{\mathbf{w}_{dn}\}_{n=1}^{N_d}$, where $\mathbb{W}_d$ is the local word subspace of $\mathcal{X}_d$. In addition, the latent variables of gene $\mathcal{X}_d$ is its topic subset $\mathcal{T}_d = \{\mathbf{t}_{dn}\}_{n=1}^{N_d}$, where $\mathbf{t}_{dn} \in \mathbb{T}_d$, and $\mathbb{T}_d$ is the local topic subspace, $\mathbb{W}_d \subseteq \mathbb{W}$, and $\mathbb{T}_d \subseteq \mathbb{T}$. Specifically, each gene shares the global topic space $\mathbb{T}_d \equiv \mathbb{T}$, where $d \in \mathbb{D}$. In this case, we suppose that each word $w \in \mathbb{W}_d$ of each gene $\mathcal{X}_d$ shares the same feature vector: $\mathbf{y}_{dw} \equiv \mathbf{y}_d = \left\{y_{df}\right\}_{f=1}^{F}$.

Then, the topic modeling process of our model can be interpreted as follows: for the training set, learning the unknown parameter $\boldsymbol{\theta}_t$, $\boldsymbol{\pi}_d$, and $\boldsymbol{\beta}_t$ from the observed variables $\mathcal{W}_d$, $\mathcal{T}_d$, and $\mathbf{y}_d$; for the testing set, predicting $\mathcal{T}_d$ and $\boldsymbol{\pi}_d$ from known parameters $\boldsymbol{\theta}_t$ and $\boldsymbol{\beta}_t$, and the observed variables $\mathcal{W}_d$ and $\mathbf{y}_d$. Obviously, $\boldsymbol{\theta}_t$ and $\boldsymbol{\beta}_t$ are global parameters, which are shared by the whole dataset. The above two steps are also called model training and predicting, and are realized by learning and inference algorithms, such as Gibbs sampling [17] and variable inference [18].

Moreover, there are two steps before model training and predicting: BoW construction and model description. Since we constructed the BoW of the gene in exactly the same way as reference [13], this step will be not repeated in this paper. For model description, there are usually two ways to describe a probabilistic graphical model, including the generative process and the graphic model, which are discussed in the next sections. The overview of our topic modeling process is depicted in Figure 2.

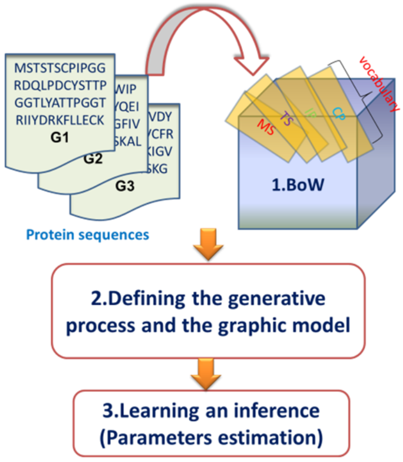


Figure 2. An overview of the topic modeling process.

2.3. Description of the Dirichlet–Multinomial Regression Latent Dirichlet Allocation Model

This section provides the description of DMR-LLDA, including its generative process and graphic model. It is worth noting that our DMR-LLDA introduces the DMR framework for gene features based on the LLDA model, so this paper emphasizes the DMR part rather than the classic LLDA.

According to DMR framework, each word sample $\mathbf{x}_{dn}$ of gene $\mathcal{X}_d$ is a "individual", and all of the samples $\{\mathbf{x}_{dn}\}_{n=1}^{N_d}$ are divided into $|\mathbb{W}_d| = W_d$ groups by word number $|\mathbb{W}_d| = W_d$. A bag of words $\{\mathbf{x}_{dn}\}_{n=1}^{N_{dw}}$ is composed by N_{dw}(the number of word w appeared in gene $\mathcal{X}_d$) samples of the w-th group, and corresponds to a feature vector $\mathbf{y}_{dw}$, which influences the latent topic $t \in \mathbb{T}$ choice of all samples.

We suppose that

$$\widetilde{\alpha}_{dwt} = \exp\left(\mathbf{y}_{dw}\boldsymbol{\beta}_t^T\right) \equiv \exp\left(\mathbf{y}_d\boldsymbol{\beta}_t^T\right) = \widetilde{\alpha}_{dt}, w \in \mathbb{W}_d \tag{2}$$

In Equation (2), $\boldsymbol{\beta}_t = \left\{\beta_{tf}\right\}_{f=1}^{F}$ represents feature parameters that correspond to topic t. Likewise, each bag of words w of gene $\mathcal{X}_d$ shares the same clustering random variable:

$$\exp(\xi_{dwt}) \equiv \exp(\xi_{dt}) = \zeta_{dt}, w \in \mathbb{W}_d \tag{3}$$

where π_{dt} is the selecting probability of the n-th word sample of gene $\mathcal{X}_d$, which chooses topic t and maximizes the utility selection U_{dnt}. In addition, $\boldsymbol{\pi}_d = \{\pi_{dt}\}_{t=1}^{T}$ is the topic weight vector of gene $\mathcal{X}_d$, which obeys the Dirichlet distribution of parameter $\left\{\delta_d^{-1}\widetilde{\alpha}_{dt}\right\}_{t=1}^{T}$:

$$p(\boldsymbol{\pi}_d) = p\left(\{\pi_{dt}\}_{t=1}^{T}\right) = \frac{\Gamma\left(\sum_{t=1}^{T}\delta_d^{-1}\widetilde{\alpha}_{dt}\right)}{\prod_{t=1}^{T}\Gamma\left(\delta_d^{-1}\widetilde{\alpha}_{dt}\right)}\prod_{t=1}^{T}\pi_{dt}^{\delta_d^{-1}\widetilde{\alpha}_{dt}-1} = \frac{\Gamma\left(\sum_{t=1}^{T}\alpha_{dt}\right)}{\prod_{t=1}^{T}\Gamma(\alpha_{dt})}\prod_{t=1}^{T}\pi_{dt}^{\alpha_{dt}-1} \tag{4}$$

where α_{dt} is the hyper-parameter of π_{dt}:

$$\alpha_{dt} = \delta_d^{-1}\widetilde{\alpha}_{dt} = \delta_d^{-1}\exp\left(\mathbf{y}_d\boldsymbol{\beta}_t^T\right) \tag{5}$$

The description of DMR-LLDA from the global and local perspective is shown below.

From the global perspective, each topic $t \in \mathbb{T}$ can be represented as a multinomial distribution over vocabulary $\mathbb{W}$, whose parameter is expressed as vector $\boldsymbol{\theta}_t = \{\theta_{tw}\}_{w=1}^{W}$, and we suppose that $\boldsymbol{\theta}_t$ obeys Dirichlet conjugate prior distribution. Each topic $t \in \mathbb{T}$ corresponds to a feature weight parameter vector $\boldsymbol{\beta}_t$, which obeys the normal distribution of parameter (μ, σ^2).

From the local perspective, each gene $\mathcal{X}_d$ is composed by N_d observed samples, which corresponds to local word number subset $\mathcal{W}_d = \{\mathbf{w}_{dn}\}_{n=1}^{N_d}$ and local latent topic number subset $\mathcal{T}_d = \{\mathbf{t}_{dn}\}_{n=1}^{N_d}$, where $\mathcal{T}_d$ obeys multinomial distribution of parameter $\boldsymbol{\pi}_d$. The local observed word subspace of gene $\mathcal{X}_d$ is $\mathbb{W}_d$, the local observed label subspace is $\mathbb{L}_d$, and the local observed feature subspace is $\mathbb{F}_d$. Each label $l \in \mathbb{L}_d$ corresponds to a topic $t \in \mathbb{T}$, where $\mathbb{T}_d \equiv \mathbb{L}_d \subseteq \mathbb{T}$ and $\boldsymbol{\Lambda}_d = \{\Lambda_{dt}\}_{t=1}^{T} = \{\Lambda_{dl}\}_{l=1}^{L}$. The dimension of topic weight $\boldsymbol{\pi}_d = \{\pi_{dt}\}_{t\in\mathbb{T}_d}$ corresponds to $\mathbb{T}_d$, which is $T_d = |\mathbb{T}_d| = |\mathbb{L}_d| \neq T$. At the same time, the range of topics on feature weight parameter vector $\boldsymbol{\beta}_t$ is limited to $t \in \mathbb{T}_d$. In addition, $\mathbf{y}_d\boldsymbol{\beta}_t^T$ decides the hyper-parameter $\boldsymbol{\alpha}_d = \{\alpha_{dt}\}_{t\in\mathbb{T}_d} = \{\alpha_{dt}\}_{t\in\mathbb{L}_d}$ of $\boldsymbol{\pi}_d$, which is the dot-product of feature vector $\mathbf{y}_d$, corresponding to feature subspace $\mathbb{F}_d$ and its weighted parameter vector $\boldsymbol{\beta}_t$.

Above all, the Dirichlet prior hyper-parameter $\boldsymbol{\alpha}_d$ of $\boldsymbol{\pi}_d$ can be expressed as

$$\boldsymbol{\alpha}_d = \{\alpha_{dl}\}_{l\in\mathbb{L}_d} = \{\alpha_{dt}\Lambda_{dt}\}_{t=1}^{T}, |\boldsymbol{\alpha}_d| = T_d = L_d \tag{6}$$

where α_{dt} is computed by Equation (5). The local topic weight $\boldsymbol{\pi}_d$ can be also represented as

$$\boldsymbol{\pi}_d = \{\pi_{dt}\}_{t\in\mathbb{T}_d} = \{\pi_{dt}\Lambda_{dt}\}_{t=1}^{T}, |\boldsymbol{\pi}_d| = T_d = L_d \tag{7}$$

Given the above, the generative process of DMR-LLDA can be described as follows. The corresponding graphical model is shown in Figure 3.

For each global topic $t \in \mathbb{T} = \{1, \ldots, T\}$, we can

(a) Generate a feature weighted parameter vector $\boldsymbol{\beta}_t = \left\{\beta_{tf}\right\}_{f=1}^{F}$ of topic t from F dimension's normal distribution of parameter (μ, σ^2):

$$\boldsymbol{\beta}_t = \left\{\beta_{tf}\right\}_{f=1}^{F} \sim N\left(\mu, \sigma^2 I\right) \tag{8}$$

(b) Generate a multinomial parameter vector $\boldsymbol{\theta}_t = \{\theta_{tw}\}_{w=1}^{W}$ from a W dimension Dirichlet distribution:

$$\boldsymbol{\theta}_t = \{\theta_{tw}\}_{w=1}^{W} \sim \mathrm{Dir}(\boldsymbol{\lambda}) \tag{9}$$

For each gene $\mathcal{X}_d$, $d \in \mathbb{D} = \{1, \ldots, D\}$. This means that

(a) We suppose that $\alpha_{dt} = \delta_d^{-1}\tilde{\alpha}_{dt}$ $(\delta_d > 0)$ as the Dirichlet prior hyper-parameter of the topic weight

$$\alpha_{dt} = \delta_d^{-1} \exp\left(\mathbf{y}_d \boldsymbol{\beta}_t^T\right) = \delta_d^{-1} \exp\left(\sum_{f=1}^{F} y_{df}\beta_{tf}\right) \tag{10}$$

(b) The binary vector $\boldsymbol{\Lambda}_d = \{\Lambda_{dt}\}_{t=1}^{T}$ limits the prior hyper-parameter $\boldsymbol{\alpha}_d$ of local topic weight

$$\boldsymbol{\alpha}_d \boldsymbol{\Lambda}_d = \{\alpha_{dt}\Lambda_{dt}\}_{t=1}^{T} \tag{11}$$

(c) We can generate local weight topic vector of topic t from a Dirichlet distribution:

$$\boldsymbol{\pi}_d = \{\pi_{dt}\Lambda_{dt}\}_{t=1}^{T} \sim \mathrm{Dir}(\boldsymbol{\alpha}_d \boldsymbol{\Lambda}_d) \tag{12}$$

(d) For each word sample $\mathbf{x}_{dn}$, we can

i. Generate topic number $\mathbf{t}_{dn}$ of $\mathbf{x}_{dn}$ from T dimensions' multinomial distribution of parameter $\boldsymbol{\pi}_d$:

$$\mathbf{t}_{dn} \sim \boldsymbol{\pi}_d \text{ or } \mathcal{T}_d = \{\mathbf{t}_{dn}\}_{n=1}^{N_d} \sim \mathrm{Mul}(\boldsymbol{\pi}_d, N_d) \tag{13}$$

ii. Generate word number $\mathbf{w}_{dn}$ of $\mathbf{x}_{dn}$ from W dimensions' multinomial distribution of parameter $\boldsymbol{\theta}_{\mathbf{t}_{dn}}$:

$$\mathbf{w}_{dn} \sim \boldsymbol{\theta}_{\mathbf{t}_{dn}} \text{ or } \mathcal{W}_d = \{\mathbf{w}_{dn}\}_{n=1}^{N_d} \sim \mathrm{Mul}\left(\boldsymbol{\theta}_{\mathbf{t}_{dn}}, N_{\mathbf{t}_{dn}}\right) \tag{14}$$

As we can see from Figure 3, $\boldsymbol{\alpha}_d$ is computed by feature vector $\mathbf{y}_d$ and its weighted parameter. Therefore, $\boldsymbol{\alpha}_d$ is a parameter rather than a random variable in the LLDA.

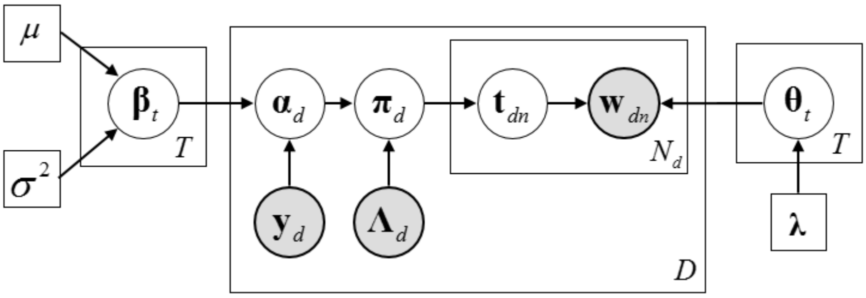


Figure 3. The graphic model of Dirichlet multinomial regression latent Dirichlet allocation (DMR-LLDA).

In our DMR-LLDA model, the unknown parameters to be estimated are the global feature parameter $\boldsymbol{\beta}$, the global topic–word multinomial distribution parameter $\boldsymbol{\theta}$, and the local topic weight $\boldsymbol{\pi}$. The hidden variable to be estimated is $\mathcal{T}$. The known data are the observed word samples $\mathcal{W}$ and binary vector $\boldsymbol{\Lambda}$. The joint distribution of $(\boldsymbol{\beta}, \boldsymbol{\pi}, \boldsymbol{\theta}, \mathcal{T}, \mathcal{W})$ is shown in Equation (15):

$$\begin{aligned} & p(\boldsymbol{\beta}, \boldsymbol{\pi}, \boldsymbol{\theta}, \mathcal{T}, \mathcal{W} | \mu, \sigma^2, \boldsymbol{\Lambda}, \boldsymbol{\lambda}) \\ & = p(\boldsymbol{\beta} | \mu, \sigma^2) \cdot \prod_{t=1}^{T} p(\boldsymbol{\theta}_t | \boldsymbol{\lambda}) \cdot \prod_{d=1}^{D} p(\boldsymbol{\pi}_d | \boldsymbol{\Lambda}_d, \boldsymbol{\alpha}_d) \prod_{n=1}^{N_d} p(\mathbf{t}_{dn} | \boldsymbol{\pi}_d) p(\mathbf{w}_{dn} | \mathbf{t}_{dn}, \boldsymbol{\theta}) \end{aligned} \tag{15}$$

Above all, the proposed method utilizes extra features as the prior knowledge of the related distribution, which is able to gain more reliable prior distribution for the LLDA; then a more precise estimation of posterior distributions is obtained.

2.4. Inference Algorithm of Dirichlet–Multinomial Regression Latent Dirichlet Allocation

The core learning task of DMR-LLDA is to compute the parameters $(\boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{\beta})$ and posterior distribution $p(\boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{\beta}, \mathcal{T} | \mathcal{W})$. The posterior estimation represents the estimating value of the parameter under the training set. The prediction process of DMR-LLDA is that on the basis of the estimated three parameters and a hidden variable, we update the unknown local parameter $\boldsymbol{\pi}$ and hidden variable of the test gene by fixing the learned global parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$; then, we get the corresponding relationship between the label and the topic. The Gibbs sampling algorithm and the variable Bayesian algorithm are two essentially approximate inference algorithms of a probabilistic graphic model, and the purpose of them is universal. In order to compare their impact on the model performance of difference inference algorithms, we designed a collapsed Gibbs sampling algorithm (CGS), a collapsed variable Bayesian algorithm (CVB), and a zero-order variational Bayesian algorithm (CVB0) for DMR-LLDA, with detail as follows.

2.4.1. The Collapsed Construction of Dirichlet–Multinomial Regression Latent Dirichlet Allocation

First of all, after the integration of model parameters $(\boldsymbol{\pi}, \boldsymbol{\theta})$ in a joint distribution, a semi-collapsed $(\boldsymbol{\beta}, \mathcal{T}, \mathcal{W})$ joint distribution is obtained:

$$\begin{aligned} & p(\boldsymbol{\beta}, \mathcal{T}, \mathcal{W} | \boldsymbol{\Lambda}, \boldsymbol{\alpha}, \boldsymbol{\lambda}, \mu, \sigma^2) = p(\boldsymbol{\beta} | \mu, \sigma^2) p(\mathcal{T} | \boldsymbol{\Lambda}, \boldsymbol{\alpha}) p(\mathcal{W} | \mathcal{T}, \boldsymbol{\lambda}) \\ & \propto \prod_{t=1}^{T} \prod_{f=1}^{F} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\beta_{tf} - \mu)^2}{2\sigma^2}} \cdot \prod_{d=1}^{D} \frac{\Gamma\left(\sum_{t=1}^{T} \alpha_{dt} \Lambda_{dt}\right)}{\Gamma\left(\sum_{t=1}^{T} \alpha_{dt} \Lambda_{dt} + N_d\right)} \prod_{t=1}^{T} \frac{\Gamma(\alpha_{dt} \Lambda_{dt} + N_{dt} \Lambda_{dt})}{\Gamma(\alpha_{dt} \Lambda_{dt})} \\ & \cdot \prod_{t=1}^{T} \frac{\Gamma\left(\sum_{w=1}^{W} \lambda_w\right)}{\Gamma\left(\sum_{w=1}^{W} \lambda_w + N_t\right)} \prod_{w=1}^{W} \frac{\Gamma(\lambda_w + N_{tw})}{\Gamma(\lambda_w)} \end{aligned} \tag{16}$$

The predictive probability distribution for the topic assignment of sample $\mathbf{x}_{dn}$ is

$$\begin{aligned} & p\left(\mathbf{t}_{dn} = t \middle| \mathcal{T}^{(\backslash dn)}, \mathcal{W}^{(\backslash dn)}, \boldsymbol{\Lambda}, \boldsymbol{\alpha}, \boldsymbol{\lambda}\right) \\ & \propto p\left(\mathbf{t}_{dn} = t, \mathbf{w}_{dn} = w \middle| \mathcal{T}^{(\backslash dn)}, \mathcal{W}^{(\backslash dn)}, \boldsymbol{\Lambda}, \boldsymbol{\alpha}, \boldsymbol{\lambda}\right) \\ & = p\left(\mathbf{t}_{dn} = t \middle| \mathcal{T}^{(\backslash dn)}, \boldsymbol{\Lambda}, \boldsymbol{\alpha}\right) p\left(\mathbf{w}_{dn} = w \middle| \mathbf{t}_{dn} = t, \mathcal{T}^{(\backslash dn)}, \mathcal{W}^{(\backslash dn)}, \boldsymbol{\lambda}\right) \\ & \propto \left(\alpha_{dt} + N_{dt}^{(\backslash dn)}\right) \Lambda_{dt} \frac{\lambda_w + N_{tw}^{(\backslash dn)}}{\sum_{w=1}^{W} \lambda_w + N_t^{(\backslash dn)}} \end{aligned} \tag{17}$$

$N_{dt}^{(\backslash dn)} \Lambda_{dt}$ is the number of samples that are assigned to the corresponding topic t of gene $\mathcal{X}_d$, except for sample $\mathbf{x}_{dn}$. $N_{tw}^{(\backslash dn)}$ is the number of samples that are assigned to the word w of topic t, except for sample $\mathbf{x}_{dn}$; therefore, $N_t^{(\backslash dn)} = \sum_{w=1}^{W} N_{tw}^{(\backslash dn)}$.

In Equation (17), α_{dt} is optimized by local observed feature vector $\mathbf{y}_d = \left\{y_{df}\right\}_{f=1}^{F}$ and global feature parameter $\boldsymbol{\beta}_t = \left\{\beta_{tf}\right\}_{f=1}^{F}$, whose updating equation is Equation (5). To simplify the updating equation, we first suppose that $\log \delta_d^{-1} = y_{df_{default}}\beta_{tf_{default}}$, and then an item of hidden global feature parameter $\beta_{tf_{default}}$ is added for global feature parameter $\boldsymbol{\beta}_t = \left\{\beta_{tf}\right\}_{f=1}^{F}$, which corresponds to a "fake" observed feature $y_{dF_{default}} = 1$. Thus, the updating equation of α_{dt} is

$$\alpha_{dt}^{\mathbf{new}} = \exp\left(\hat{\mathbf{y}}_d \hat{\boldsymbol{\beta}}_t^T\right) = \exp\left(y_{df_{default}}\beta_{tf_{default}} + \sum_{f=1}^{F} y_{df}\beta_{tf}\right) \tag{18}$$

$$\begin{gathered} \hat{\boldsymbol{\beta}}_t = \left\{\boldsymbol{\beta}_t, \beta_{tF_{default}}\right\} = \left\{\beta_{t1}, \beta_{t2}, \ldots, \beta_{tF}, \beta_{tF_{default}}\right\} \\ \hat{\mathbf{y}}_d = \left\{\mathbf{y}_d, y_{dF_{default}}\right\} = \{y_{d1}, y_{d2}, \ldots, y_{dF}, 1\} \end{gathered} \tag{19}$$

2.4.2. The Optimization of the Feature Parameters of Dirichlet–Multinomial Regression Latent Dirichlet Allocation

For Gibbs sampling or variable Bayesian, we need to update the global feature parameter $\hat{\boldsymbol{\beta}}_t$ in the inference process. We adopted the method of gradient descent for optimizing $\hat{\boldsymbol{\beta}}_t$.

In Equation (16), the $\hat{\boldsymbol{\beta}}$-related section is

$$F(\hat{\boldsymbol{\beta}}) \propto \prod_{t=1}^{T}\prod_{f=1}^{F+1} e^{-\frac{(\beta_{tf}-\mu)^2}{2\sigma^2}} \cdot \prod_{d=1}^{D} \frac{\Gamma\left(\sum_{t=1}^{T} \alpha_{dt}\Lambda_{dt}\right)}{\Gamma\left(\sum_{t=1}^{T} \alpha_{dt}\Lambda_{dt} + N_d\right)} \prod_{t=1}^{T} \frac{\Gamma(\alpha_{dt}\Lambda_{dt} + N_{dt}\Lambda_{dt})}{\Gamma(\alpha_{dt}\Lambda_{dt})} \tag{20}$$

Based on the logarithm of Equation (20), we take the derivative with respect to global feature parameter β_{tf} and adjust it to zero. The updated equation of β_{tf} is

$$\begin{aligned} \beta_{tf}^{\mathbf{new}} = \sigma^2 \sum_{d=1}^{D} y_{df}\alpha_{dt}^{\mathbf{new}}\Lambda_{dt}\{ \quad & \Psi\left(\textstyle\sum_{t=1}^{T} \alpha_{dt}^{\mathbf{new}}\Lambda_{dt}\right) - \Psi\left(\textstyle\sum_{t=1}^{T} \alpha_{dt}^{\mathbf{new}}\Lambda_{dt} + N_d\right) \\ & + \Psi\left(\alpha_{dt}^{\mathbf{new}}\Lambda_{dt} + N_{dt}\Lambda_{dt}\right) - \Psi\left(\alpha_{dt}^{\mathbf{new}}\Lambda_{dt}\right)\} + \mu \\ t \in \mathbb{T} = \{1, \ldots, T\} \qquad\qquad & f \in \mathbb{F}' = \{1, \ldots, F, F+1\} \end{aligned} \tag{21}$$

Finally, $\alpha_{dt}^{\mathbf{new}}$ is updated by Equation (18).

2.4.3. The Collapsed Gibbs Sampling Algorithm of the Dirichlet–Multinomial Regression Latent Dirichlet Allocation

To determine the initial state of the Markov chain, we initiate the hidden topic number $\mathbf{t}_{dn}$ of each sample $\mathbf{x}_{dn}$ first; then, we utilize the predictive probability of hidden variable $\mathbf{t}_{dn}$ from Equation (17) as the state transition probability of the Markov chain. In the process of Gibbs sampling, the topic number $\mathbf{t}_{dn}$ of each sample $\mathbf{x}_{dn}$ is updated, and the hyper-parameter $\boldsymbol{\alpha}_d = \{\alpha_{dt}\Lambda_{dt}\}_{t=1}^{T}$ is also updated by Equation (18). Finally, the global feature parameter β_{tf} is updated by Equation (21).

After several iterations in the burn-in time, the Markov chain is attracted to objective distribution, and then the posterior distribution $p\left(\boldsymbol{\beta}, \mathcal{T} \middle| \mathcal{W}, \mu, \sigma^2, \boldsymbol{\alpha}, \boldsymbol{\lambda}\right)$ is estimated. The posterior estimation of the local topic weight $\boldsymbol{\pi}_d = \{\pi_{dt}\Lambda_{dt}\}_{t=1}^{T}$ and topic–word multinomial distribution parameter is

$$\hat{\pi}_{dt} = \frac{\alpha_{dt}\Lambda_{dt} + \mathbb{E}[N_{dt}\Lambda_{dt}]}{\sum_{t=1}^{T}(\alpha_{dt}\Lambda_{dt} + \mathbb{E}[N_{dt}\Lambda_{dt}])} \tag{22}$$

$$\hat{\theta}_{tw} = \frac{\lambda_w + \mathbb{E}[N_{tw}]}{\sum_{w=1}^{W}(\lambda_w + \mathbb{E}[N_{tw}])} \tag{23}$$

2.4.4. The Collapsed Variable Bayesian Inference Algorithm of the Dirichlet–Multinomial Regression Latent Dirichlet Allocation

The whole variational objective function before being collapsed is

$$\begin{aligned} F(\boldsymbol{\eta}) &= \mathbb{E}_q[\log p(\boldsymbol{\beta},\boldsymbol{\pi},\boldsymbol{\theta},\mathcal{T},\mathcal{W})] - \mathbb{E}_q[\log q(\boldsymbol{\beta},\boldsymbol{\pi},\boldsymbol{\theta},\mathcal{T}|\boldsymbol{\eta})] \\ &= \mathbb{E}_q[\log p(\mathcal{T},\mathcal{W})] - \mathbb{E}_q[\log q(\mathcal{T}|\boldsymbol{\eta})] \\ &= \mathrm{KL}(q(\mathcal{T}|\boldsymbol{\eta})||p(\mathcal{T},\mathcal{W})) \end{aligned} \tag{24}$$

After margining the model parameters $(\boldsymbol{\pi},\boldsymbol{\theta})$, the objective function is

$$\begin{aligned} \mathcal{F} = \; & \mathbb{E}_{q(\mathbf{t}_{dn})}\left[\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[\log p\left(\mathbf{t}_{dn},\mathbf{w}_{dn}\middle|\mathcal{T}^{(\backslash dn)},\mathcal{W}^{(\backslash dn)}\right)\right]\right] \\ & -\mathbb{E}_{q(\mathbf{t}_{dn})}[\log q(\mathbf{t}_{dn})] + Const_{q(\mathbf{t}_{dn})} \end{aligned} \tag{25}$$

where $Const_{q(\mathbf{t}_{dn})}$ represents the unrelated item with variational distribution $q(\mathbf{t}_{dn})$. There are two kinds of construction below:

$$\mathcal{F} = \mathrm{KL}\left(q(\mathbf{t}_{dn})\middle|\middle|\exp\left\{\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[\log p\left(\mathbf{t}_{dn},\mathbf{w}_{dn}\middle|\mathcal{T}^{(\backslash dn)},\mathcal{W}^{(\backslash dn)}\right)\right]\right\}\right) + Const_{q(\mathbf{t}_{dn})} \tag{26}$$

$$\mathcal{F} \geq \mathrm{KL}\left(q(\mathbf{t}_{dn})\middle|\middle|\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[p\left(\mathbf{t}_{dn},\mathbf{w}_{dn}\middle|\mathcal{T}^{(\backslash dn)},\mathcal{W}^{(\backslash dn)}\right)\right]\right) + Const_{q(\mathbf{t}_{dn})} \tag{27}$$

In Equation (26), the updating equation of optimal variational parameter η^*_{dnt} by a CVB algorithm is

$$\begin{aligned} \eta^*_{dnt} = \; & q^*_{\mathrm{CVB}}(\mathbf{t}_{dn}=t) \approx \exp\left\{\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[\log\left(\alpha_{dt}+N^{(\backslash dn)}_{dt}\right)\Lambda_{dt}\right]\right\} \\ & +\exp\left\{\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[\log\left(\lambda_w+N^{(\backslash dn)}_{tw}\right)\right]\right\} - \exp\left\{\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[\log\left(\textstyle\sum_{w=1}^{W}\lambda_w+N^{(\backslash dn)}_{t}\right)\right]\right\} \end{aligned} \tag{28}$$

Each expectation of the above equation is

$$\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[\log\left(\alpha_{dt}+N^{(\backslash dw)}_{dt}\right)\Lambda_{dt}\right] = \log\left(\alpha_{dt}\Lambda_{dt}+\mu_{N^{(\backslash dw)}_{dt}}\right) - \frac{\sigma^2_{N^{(\backslash dw)}_{dt}}}{2\left(\alpha_{dt}\Lambda_{dt}+\mu_{N^{(\backslash dw)}_{dt}}\right)^2} \tag{29}$$

$$\left\{\begin{array}{lll} \mu_{N^{(\backslash dw)}_{dt}} = \mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[N^{(\backslash dw)}_{dt}\Lambda_{dt}\right] & = & \sum\limits_{w=1}^{W}(N_{dw}-1)\eta_{dwt}\Lambda_{dt} \\ \sigma^2_{N^{(\backslash dw)}_{dt}} = \mathbb{V}_{q(\mathcal{T}^{(\backslash dn)})}\left[N^{(\backslash dw)}_{dt}\Lambda_{dt}\right] & = & \sum\limits_{w=1}^{W}(N_{dw}-1)\eta_{dwt}\Lambda_{dt}\left(1-\sum_{w=1}^{W}(N_{dw}-1)\eta_{dwt}\Lambda_{dt}\right) \end{array}\right. \tag{30}$$

$$\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[\log\left(\lambda_w+N^{(\backslash dw)}_{tw}\right)\right] = \log\left(\lambda_w+\mu_{N^{(\backslash dw)}_{tw}}\right) - \frac{\sigma^2_{N^{(\backslash dw)}_{tw}}}{2\left(\lambda_w+\mu_{N^{(\backslash dw)}_{tw}}\right)^2} \tag{31}$$

$$\left\{\begin{array}{lll} \mu_{N^{(\backslash dw)}_{tw}} = \mathbb{E}_{q(\mathcal{T}^{(\backslash dw)})}\left[N^{(\backslash dw)}_{tw}\right] & = & \sum\limits_{d=1}^{D}(N_{dw}-1)\eta_{dwt}\Lambda_{dt} \\ \sigma^2_{N^{(\backslash dw)}_{tw}} = \mathbb{V}_{q(\mathcal{T}^{(\backslash dw)})}\left[N^{(\backslash dw)}_{tw}\right] & = & \sum\limits_{d=1}^{D}(N_{dw}-1)\eta_{dwt}\Lambda_{dt}(1-(N_{dw}-1)\eta_{dwt})\Lambda_{dt} \end{array}\right. \tag{32}$$

$$\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[\log\left(\sum_{w=1}^{W}\lambda_w+N^{(\backslash dw)}_{t}\right)\right] = \log\left(\sum_{w=1}^{W}\lambda_w+\mu_{N^{(\backslash dw)}_{t}}\right) - \frac{\sigma^2_{N^{(\backslash dw)}_{t}}}{2\left(\sum_{w=1}^{W}\lambda_w+\mu_{N^{(\backslash dw)}_{t}}\right)^2} \tag{33}$$

$$\begin{cases} \mu_{N_t^{(\backslash dw)}} = \mathbb{E}_{q(\mathcal{T}^{(\backslash dw)})}\left[N_t^{(\backslash dw)}\right] = \sum\limits_{d=1}^{D}\sum\limits_{w=1}^{W}(N_{dw}-1)\eta_{dwt}\Lambda_{dt} \\ \sigma^2_{N_t^{(\backslash dw)}} = \mathbb{V}_{q(\mathcal{T}^{(\backslash dw)})}\left[N_t^{(\backslash dw)}\right] = \sum\limits_{d=1}^{D}\sum\limits_{w=1}^{W}(N_{dw}-1)\eta_{dwt}\Lambda_{dt}(1-(N_{dw}-1)\eta_{dwt}\Lambda_{dt}) \end{cases} \tag{34}$$

In Equation (27), the updating equation of the optimal variational parameter η^*_{dnt} by CVB0 algorithm is

$$\eta^*_{dnt} = q^*_{\text{CVB0}}(\mathbf{t}_{dn} = t) \approx \left(\alpha_{dt}\Lambda_{dt} + \mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[N_{dt}^{(\backslash dn)}\Lambda_{dt}\right]\right)\frac{\lambda_w + \mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[N_{tw}^{(\backslash dn)}\right]}{\sum_{w=1}^{W}\lambda_w + \mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[N_t^{(\backslash dn)}\right]} \tag{35}$$

The plenitude statistic of samples in Equation (35) are $N_{dt}^{(\backslash dn)}$,$N_{tw}^{(\backslash dn)}$, and $N_t^{(\backslash dn)}$, and their expectation under variational distribution $q\left(\mathcal{T}^{(\backslash dn)}\right)$ is

$$\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[N_{dt}^{(\backslash dn)}\Lambda_{dt}\right] = \sum_{i=1,i\neq n}^{N_d}\mathbb{I}(\mathbf{t}_{dn}=t)\eta_{dit}\Lambda_{dt} \tag{36}$$

$$\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[N_{tw}^{(\backslash dn)}\right] = \sum_{d=1}^{D}\sum_{i=1,i\neq n}^{N_d}\mathbb{I}(\mathbf{t}_{dn}=t)\mathbb{I}(\mathbf{w}_{dn}=w)\eta_{dit}\Lambda_{dt} \tag{37}$$

$$\mathbb{E}_{q(\mathcal{T}^{(\backslash dn)})}\left[N_t^{(\backslash dn)}\right] = \sum_{d=1}^{D}\sum_{i=1,i\neq n}^{N_d}N_{di}\eta_{dit}\Lambda_{dt} \tag{38}$$

The $\backslash dn$ in the above equation can be adapted to $\backslash dw$, because the bag of words dw not only shares a similar word number $\mathbf{w}_{dn} = w$, but also shares the same topic number $\mathbf{t}_{dn} = t$. Then, optimal variational distribution η^*_{dnt} can be adapted to η^*_{dwt}.

$$\gamma_{dt} = \mathbb{E}_{q(\mathcal{T}^{(\backslash dw)})}\left[N_{dt}^{(\backslash dw)}\Lambda_{dt}\right] = \sum_{w=1}^{W}(N_{dw}-1)\eta_{dwt}\Lambda_{dt} \tag{39}$$

$$\mu_{tw} = \mathbb{E}_{q(\mathcal{T}^{(\backslash dw)})}\left[N_{tw}^{(\backslash dw)}\right] = \sum_{d=1}^{D}(N_{dw}-1)\eta_{dwt}\Lambda_{dt} \tag{40}$$

$$\mu_t = \mathbb{E}_{q(\mathcal{T}^{(\backslash dw)})}\left[N_t^{(\backslash dw)}\right] = \sum_{d=1}^{D}\sum_{w=1}^{W}(N_{dw}-1)\eta_{dwt}\Lambda_{dt} \tag{41}$$

The inference equation difference between CVB and CVB0 shows that CVB only retains the zero-order information of the Taylor expansion; however, CVB0 is the re-collapse of a hidden variable space based on Jensen inequality. Therefore, CVB0 is much more precise than CVB. The corresponding algorithm of CVB and CVB0 are shown in Tables 1 and 2 respectively.

Table 1. Collapsed variable Bayesian (CVB) algorithm of DMR-LLDA.

```
CVB algorithm of DMR-LLDA
Initialize global variational parameters
While the number of iterations $r < r_{\max}$ or $F$ is not converged do
    For $d = 1 : D$    do
        Initialize local variational parameters to constant
        Repeat (the local variational inference of gene $d$)
            $\eta_{dwt}^{(r)} \propto \frac{\left(\Lambda_{dt}\alpha_{dt}+\mu_{N_{dt}}^{(r-1)}\right)\left(\lambda_w+\mu_{N_{tw}}^{(r-1)}\right)}{\left(\sum_{w=1}^{W}\lambda_w+\mu_{N_t}^{(r-1)}\right)} e^{-\frac{(\sigma_{N_{dt}}^{(r-1)})^2}{2(\alpha_t+\mu_{N_{dt}}^{(r-1)})^2}} e^{-\frac{(\sigma_{N_{tw}}^{(r-1)})^2}{2(\lambda_w+\mu_{N_{tw}}^{(r-1)})^2}} e^{-\frac{(\sigma_{N_t}^{(r-1)})^2}{2(\sum_{w=1}^{W}\lambda_w+\mu_{N_t}^{(r-1)})^2}}$
            Update $\mu_{N_{dt}}^{(r)}$ and $\sigma_{N_{dt}}^{(r)}$ by Equations (29)~(30)
            $\alpha_{dt}^{(r)} = \exp\left(\hat{\mathbf{y}}_d \hat{\boldsymbol{\beta}}_t^{(r-1)}\right)$
        Until $\gamma_{dt}^{(r)}$ is converged: $(1/N_d)\sum_{t=1}^{T}\left|\gamma_{dt}^{(r)}-\gamma_{dt}^{(r-1)}\right| < 0.00001$
    End For
    $\mu_{N_{tw}}^{(r)}$, $\sigma_{N_{tw}}^{(r)}$, $\mu_{N_t}^{(r)}$ and $\sigma_{N_t}^{(r)}$ by Equations (31)~(34)
    Update $\beta_{tf}^{(r)}$ by Equation (21)
End while
```

Table 2. Zero-order variational Bayesian (CVB0) algorithm of DMR-LLDA.

```
CVB0 algorithm of DMR-LLDA
Initialize global variational parameters
While the number of iterations $r < r_{\max}$ or $\mathcal{F}$ is not converged do
    For $d \in \mathbb{D}$ do
        Initialize local variational parameters to constant
        Repeat: (the local variational inference of gene $d$)
            $\eta_{dwt}^{(r)} \propto \left(\alpha_{dt}\Lambda_{dt} + \gamma_{dt}^{(r-1)}\right)\frac{\lambda_w+\mu_{tw}^{(r-1)}}{\sum_{w=1}^{W}\lambda_w+\mu_t^{(r-1)}}$
            $\gamma_{dt}^{(r)} = \sum_{w=1}^{W}(N_{dw}-1)\eta_{dwt}^{(r)}$
            $\alpha_{dt}^{(r)} = \exp\left(\hat{\mathbf{y}}_d \hat{\boldsymbol{\beta}}_t^{(r-1)}\right)$
        Until $\gamma_{dt}^{(r)}$ is converged: $(1/N_d)\sum_{t=1}^{T}\left|\gamma_{dt}^{(r)}-\gamma_{dt}^{(r-1)}\right| < 0.00001$
    End For
    $\mu_{tw}^{(r)} = \sum_{d=1}^{D}(N_{dw}-1)\eta_{dwt}^{(r)}$, $\mu_t^{(r)} = \sum_{d=1}^{D}\sum_{w=1}^{W}(N_{dw}-1)\eta_{dwt}^{(r)}$
    Update $\beta_{tf}^{(r)}$ by Equation (21)
End while
```

3. Materials and Results

This section provides a concise and precise description of the experimental results, their interpretation, and the experimental conclusions that can be drawn.

3.1. Dataset

In this paper, the validity and accuracy of proposed models are tested on the S.cerevisiae (S.C) dataset, which is introduced in reference [12]. This dataset includes several aspects of the yeast genome, such as sequence statistics, phenotype, expression, secondary structure, and homology. Meanwhile, two kinds of function annotation standard, including FunCat and GO, are used to annotate gene function. Due the universality of GO, the dataset depends on the GO that is adopted in our experiments. As described in Section 2.1, the construction of the BoW is based on amino acid composition, so we mainly use one of datasets that depends on the sequence statistics. In addition, we construct a dataset named S.C-CC from S.C, which only includes the GO terms belonging to the cellular component (CC). Therefore, there are fewer GO terms in the S.C-CC dataset when compared with the S.C dataset, and both of them are used in our experiments for investigating the influence of different label numbers on prediction performance. The statistics of the S.C and S.C-CC dataset is shown in Table 3. In this set, F denotes the number of GO terms, D denotes the number of genes, and W denotes the size of the vocabulary.

Table 3. The statistics of the S.cerevisiae (S.C) and S.cerevisiae-cellular component (S.C-CC) datasets.

Dataset	D	W	F	L
S.C	1692	400	4133	1538
S.C-CC			547	319

As shown in Table 3, there are 1692 genes and 4133 function labels in the S.C dataset; in the S.C-CC dataset, there are 1692 genes and 547 function labels. Due to the large number of GO terms in the gene function dataset, we utilized a Boolean matrix decomposition (BMD) method to reduce the dimensionality of the function labels. BMD is a kind of label space dimension reduction (LSDR) method [19], which addresses the multi-label classification problem with many labels. LSDR approaches use a compression step to transform the original high dimension label space into a lower dimension label space, and then multi-label classifiers are trained on a dataset with fewer labels, which can reduce the computation burden of the classifier. The existing studies about LSDR show that LSDR approaches are useful for optimizing the running time and accuracy of multi-label classification. In our BMD process, original label matrix $Y \in \{0,1\}^{D\times F}$ (D denotes the number of genes, and F denotes the number of features) is decomposed into the product of two matrices, $C \in \{0,1\}^{D\times L}$ (L denotes the number of labels) and $B \in \{0,1\}^{L\times F}$, where $Y = C \circ B$ ($\circ$ denotes Boolean product) is satisfied. We also called it exact BMD and adopted this algorithm, which is proposed in reference [20]. Compared with other LSDR algorithms, an exact BMD can retain the interpretability of low dimension label space and restore the low dimension-predicted label matrix to the original label matrix by matrix B. At last, the number of function labels is reduced into a smaller dimension, and L denotes the number of GO terms after label space dimension reducing. Then DMR-LLDA actually needs to process 1358 GO terms of the S.C dataset and 319 GO terms of the S.C-CC dataset. Nonetheless, the lower dimensional label space can be recovered by a Boolean product after predicting, so we still get the whole function labels sets in the prediction results.

DMR-LLDA's advantage here over LLDA lies in the introduction of extra features. In the S.C and S.C-CC dataset, there are six extra gene features for each gene, including the molecular weight of the gene, the isoelectric point, the average coefficients of hydrophilic, the number of exons, the adaptability index of the codon, the number of motifs, and the open reading frame (ORF) number of chromosomes. The statistics of extra features are shown in Table 4.

Table 4. The statistics of extra features in the S.C dataset.

Feature Name	Notation	Type
molecular weight	mol_wt	Integer
isoelectric point	theo_pI	Real numbers
average coefficients of hydrophilic	hydro	Real numbers
number of exons	position	Integer
adaptability index of codon	Cai	Real numbers
number of motifs	motifs	Integer
ORF number of chromosomes	chromosome	Integer

ORF: Open reading frame

As the max word length of the S.C dataset is two amino acid alphabets (G,A,V,L,I,F,P, Y,S,C,M,N,Q,T,D,E,K,R,H,W), a human dataset constructed by ourselves is adopted to evaluate the impact on the topic model performance of word length. This human dataset is constructed in a similar way as in reference [13]. In addition, we also constructed two human datasets for different word lengths, where the max word length of the Human1 dataset is two amino acid alphabets, and that of the Human2 dataset is three amino acid alphabets. For the Human2 dataset, the original number of words is 8400, but we filtered several words that have a high frequency. Then, the statistic of the S.C and S.C-CC dataset is shown in Table 5.

Table 5. The statistic of two human datasets.

Dataset	D	W	L
Human1	4962	5297	1477
Human2		400	

3.2. Parameter Settings and Evaluation Criterias

The DMR-LLDA learning framework involves four different parameters: μ, σ^2, α, and λ. The α and λ are the parameters of the two-Dirichlet distribution, where the larger the value of λ, the more balanced the probability of a word in a topic. The setting of the λ value has been discussed in reference [13]. Nonetheless, the value of α is optimized by a protein feature, so its initial value does not have a big effect on model performance. According to the experience, we set $\alpha = 50/T$ as the initial value, and set $\lambda = 200/W$, with $T = L$. In addition, μ and σ^2 are respectively the mean and variance of normal distribution, obeyed by feature weighted parameter $\boldsymbol{\beta}$, and we set $\mu = 0, \sigma^2 = 1$.

In the Gibbs sampling process of model training, we set the number of the Markov chain as 1 and the maximum number of iterations is 2000 times, where the number of iterations of burn-in time is set to 1000. We record the state space at intervals of 50 times on the converged Markov chain, and 20 times per record is conducted. In the process of model predicting, we set the number of iterations as 1000 times. After 500 iterations for the burn-in time, we record the state space at intervals of 50 times. In the variable Bayesian inferring process of model training, we initialize the global variable parameter μ_{tw} through random number s and hyper-parameter λ_w: $\mu_{tw} = \lambda_w + (s * \lambda_w)/10$; in each local variable inference, we set the converged threshold as 0.00001, and the maximum number of times of the local variable inference as 100. The number of global scanning iterations is 1000.

The five-fold cross validation is conducted to measure and compare the performance of DMR-LLDA and the comparative algorithms. Five representative multi-label learning evaluation criteria are used in this paper, including hamming loss (HL), average precision (AP), one error, and micro-averaged and macro-averaged F1 scores (Micro-F1 and Macro-F1). In addition, three kinds of areas under a precision–recall curve are also used, including $\overline{AUPRC}$, $AU(\overline{PRC})$, and $\overline{AUPRCw}$, which is proposed in reference [12]. Finally, we repeat the random partition and evaluation in five independent rounds, and report the average results.

3.3. The Impact of Word Length on Model Performance

Firstly, the performance comparison of the LLDA model between the Human1 and Human2 datasets is shown in Figure 4. As shown in Figure 4, we find that the value of $\overline{AUPRC}$ and $\overline{AUPRCw}$ in Human1 is higher than that in Human2; the value of the AP on Human1 is lower than that of Human2; and the value of one error, HL, and $AU(\overline{PRC})$ is almost equal to that of Human1 and Human2. These results show that the classification performance of the LLDA on Human1 and Human2 is almost the same, which reveals that the larger word space might not obtain a better classifying performance.

Moreover, related studies suggested that a word length of more than four amino acid alphabets would not improve the classification accuracy, and would only increase the complexity of computation [21]. Therefore, in the following experiments, we only adopt the S.C and S.C-CC datasets whose word length is two amino acid alphabets.

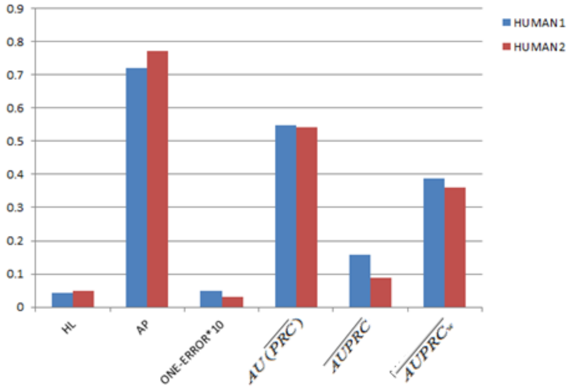


Figure 4. The comparisons between the Human1 and Human2 datasets. Define terms if necessary.

3.4. Gene Function Prediction with Cross Validation

In addition to LLDA, we also adopted three widely adopted methods: multi-label *k*-nearest neighbor (MLKNN) [22], back propagation for multi-label learning (BPMLL) [23], and support vector machines (SVMs) for performance comparison. MLKNN and BPMLL are two representative multi-label classifiers, and can be performed by an open source tool called Mulan [24]. SVMs adopt a "one-versus-all" scheme, which trains each label by a binary SVM independently and is implemented using the LibLinear software package [25]. These five models are trained and used to predict with the S.C and S.C-CC datasets. Figure 5 shows the HL, AP, one error, Micro-F1, Macro-F1, $AU(\overline{PRC})$, $\overline{AUPRC}$, and $\overline{AUPRCw}$ values of all models in the two datasets, respectively. For AP, Micro-F1, Macro-F1, $AU(\overline{PRC})$, $\overline{AUPRC}$, and $\overline{AUPRCw}$, the larger the value, the better the performance. Conversely, for HL and one error, the smaller the value, the better the performance. The red asterisk of Figure 5 represents the best results in each dataset. It is worth noting that the experimental results of this section are obtained by a CGS inference algorithm.

As shown in Figure 5, DMR-LLDA can achieve better results in almost all evaluation criteria for the two datasets. The concrete analysis is introduced as follows.

For the S.C dataset, the DMR-LLDA achieves the best performance for AP, $AU(\overline{PRC})$, $\overline{AUPRC}$, $\overline{AUPRCw}$, Micro-F1, and Macro-F1. For example, with AP, the DMR-LLDA achieves 94%, 3.3%, 96%, and 26% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. With $AU(\overline{PRC})$, the DMR-LLDA achieves 109%, 2.3%, 89%, and 24% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. For $\overline{AUPRC}$, the DMR-LLDA achieves 31%, 39%, 44%, and 25% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. For $\overline{AUPRCw}$, the DMR-LLDA achieves 33%, 8.1%, 48%, and 10% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. With Micro-F1, the DMR-LLDA achieves 116%, 6.1%, 123%, and 29% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. On Macro-F1, DMR-LLDA achieves 22%, 2.9%, 24%, and 25% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. Nevertheless, for one error and HL, SVMs get better results than the DMR-LLDA.

For the S.C-CC dataset, the DMR-LLDA obtains a better performance in terms of AP, $AU(\overline{PRC})$, $\overline{AUPRC}$, $\overline{AUPRCw}$, Micro-F1, and Macro-F1. For AP, the DMR-LLDA achieves 36%, 1.7%, 39%, and 30% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. For $AU(\overline{PRC})$, the DMR-LLDA achieves 68%, 4%, 64%, and 20% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. For $\overline{AUPRC}$, the DMR-LLDA achieves 73%, 35%, 62%, and 34% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. For $\overline{AUPRCw}$, the DMR-LLDA achieves 67%, 6.4%, 92%, and 23% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. For Micro-F1, the DMR-LLDA achieves 101%, 4.1%, 114%, and 26% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. For Macro-F1, the DMR-LLDA achieves 18%, 1.8%, 16%, and 20% improvements over MLKNN, LLDA, BPMLL, and SVMs, respectively. Nevertheless, for one error, BPMLL gets better results than the DMR-LLDA; for HL, SVMs gets better results than the DMR-LLDA.

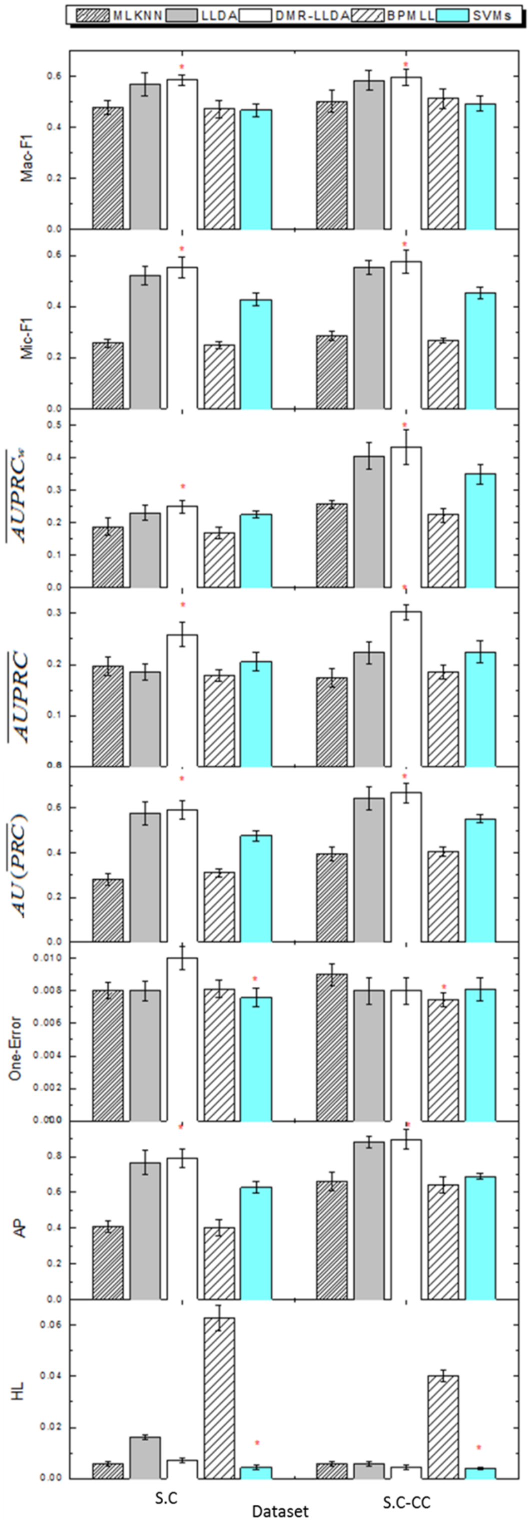


Figure 5. The comparisons between DMR-LLDA, LLDA, back propagation for multi-label learning (BPMLL), support vector machines (SVMs), and multi-label *k*-nearest neighbor (MLKNN) in two datasets. The red asterisk represents the best results in each dataset.

For both of the datasets, we can find that the improvements on $\overline{AUPRC}$ are more significant than $AU(\overline{PRC})$ and $\overline{AUPRCw}$, which indicates that the DMR-LLDA has a stronger effect on improving the overall accuracy of gene function prediction without respect to label weights. In the comparisons of the S.C and S.C-CC datasets, we find that the values of AP, $AU(\overline{PRC})$, $\overline{AUPRC}$, and $\overline{AUPRCw}$ in the S.C dataset is lower than in the S.C-CC dataset, and the value of one error and HL in the S.C is higher than in the S.C-CC dataset. This is due to the same word space and different label number between these two datasets. The fewer labels of the S.C-CC dataset can promote a higher classifying performance.

Above all, these results indicate that the DMR-LLDA can further improve the accuracy of gene function prediction by introducing the DMR framework into the LLDA model, which optimizes the hyper-parameters of the topic weight. Meanwhile, the DMR-LLDA has an apparent advantage in improving the overall prediction accuracy.

3.5. The Impact on Prior Parameters of Feature Variables

In the DMR-LLDA model, the introduction of gene features is realized by feature weight parameter $\boldsymbol{\beta}$. Then the operation on topic parameter vector $\boldsymbol{\beta}_t$ and feature vector $\mathbf{y}_d$ are reflected in Dirichlet hyper parameter $\boldsymbol{\alpha}_d$. Table 6 shows the impact of different feature values on prior parameters $\boldsymbol{\alpha}_d$.

Table 6. The impact on prior parameters of feature variables.

$\boldsymbol{\alpha}_d$	The words under topic when mol_wt = 49629.3, theo_pI = 8.96, hydro = 0.1, position = 1, Cai = 0.17, motifs = 2, chromosome = 16
1.88	GM IH LH VH LK IG GC IC AK VM FG AM LW IK VG VW FC IG FH GK
1.32	LM ST SM LT KM LL IM KL LF SL EM LP DM IT LK EF KT LE SK SP
0.79	GH VC AC KC GC AL GM LH AH AF AM VW AW GW EC KH TH GF AT GT
0.64	IL VG GE FM YK QW YM VW GP TL KT LW RP LQ IR FH NW NX FS PM
0.23	TT SV TV ST SW SQ TQ PT PV IV SP TM CT QT AV TP TC SC VV NV
$\boldsymbol{\alpha}_d$	The words under topic when mol_wt = 85873.7, theo_pI = 9.74, hydro=0.664 position = 1, Cai = 0.1, motifs = 2, chromosome = 16
4.23	KR TF KE QS LW EW DM YF QT SM LX SF IN QW LR VL VS QG MC QC
3.77	LM SM LS RC DW EM LE QT LV EW FM QI RM NE DT IE FT AR QC GP
0.23	QM KR AP EF LF QR HP EC RE RF DS VE EW KF FE LT TL QV QC AR
0.11	CF PI ED QY GQ HN RI HD HI SN YQ TQ PW RH YL PQ PN SI QE RS
0.09	SW VF NW AC DF TW EQ LW EH MC DM AW PS GV VQ AQ ID TG RF VE

For the LLDA, the hyper-parameter value is set as a fixed value. However, Table 6 shows that only the different values on mol_wt, theo_pI, hydro, and Cai make a significant difference of hyper-parameter value in the DMR-LLDA, which is also the main way for gene features to impact label allocation.

3.6. The Comparison Results of Inference Algorithms

We designed three kinds of inference algorithm for the DMR-LLDA, including CGS, CVB, and CVB0. This section compares CGS with CVB0 in the S.C dataset. The experimental results are shown in Figure 6. As shown in Figure 6, the overall performance of CVB0 is better than the performance of CGS. Concrete analysis is represented as follows.

For the S.C dataset, CVB0 achieves the best results in AP, $\overline{AUPRC}$, $AU(\overline{PRC})$, and $\overline{AUPRCw}$, and achieves almost similar results in HL. However, CVB0 has a worse value in one error.

For the S.C-CC dataset, CVB0 achieves the best results in AP, one error, $\overline{AUPRC}$, $AU(\overline{PRC})$, and $\overline{AUPRCw}$, and achieves almost similar results in HL. The above results demonstrate the validity of the designed inference algorithms for the DMR-LLDA. Meanwhile, the experimental results indicate that the CVB0 inference algorithm can obtain more precise prediction results by the re-collapse of hidden variable space based on Jensen inequality.

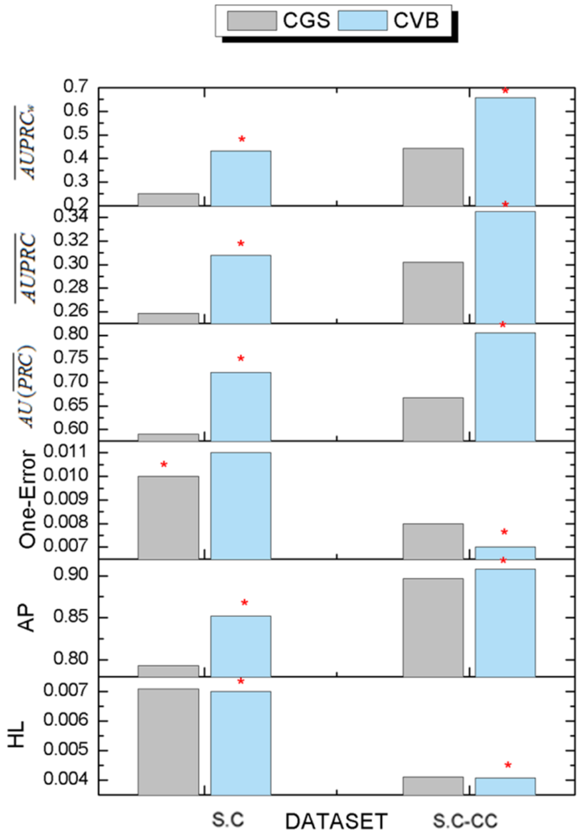


Figure 6. The comparison results with CVB0 and CGS. The red asterisk represents the best results in each dataset.

Above all, due to the lack of prior knowledge, the prior distributions of the Bayesian model are usually set for convenience. Meanwhile, the parameters of prior distribution are also set as a fixed value based on experience, which makes the inaccurate estimation of posterior distributions. In our DMR-LLDA model, the gene feature information is introduced into the LLDA as the prior knowledge by the DMR framework. The hyper-parameter of the prior distribution is updated in the inference process rather than by a fixed constant, which can improve the estimation precision of posterior distributions, so as to improve the accuracy of gene function prediction.

4. Conclusions

In this paper, we introduce multiple types of features into gene function prediction based on a multi-label surprised topic model, and propose a multi-label supervised topic model conditioned on arbitrary features named the DMR-LLDA. By applying an exponential a priori constructed previously with weighted features on the hyper-parameters of gene-topic (or label) distribution, this model can utilize the observed features of each gene in multi-label topic modeling. Furthermore, three learning algorithms are designed for this model, including CGS, CVB inference, and CVB0 inference. The predictive performance of this model is measured by the AP, one error, Hamming loss, $\overline{AUPRC}$, $AU(\overline{PRC})$, $\overline{AUPRCw}$, Micro-F1, and Macro-F1. Experiments on a standard dataset show that the DMR-LLDA is superior to the LLDA, MLKNN, BPMLL, and SVM models. Meanwhile, experimental results show that the DMR-LLDA can get a much more accurate estimation of posterior distribution, due to using the gene feature information in addition to the amino acid sequence.

Author Contributions: Conceptualization: L.L. and W.Z.; methodology: L.L.; software: L.L., L.T., and X.J.; validation: L.L. and L.T.; formal analysis: L.L.; writing (original draft preparation): L.L. and L.T.; writing (review and editing): L.T. and X.J.

Funding: This research was funded by the National Natural Science Foundation of China (grant number 61862067) and the Doctor Science Foundation of Yunnan Normal University (no. 01000205020503090, no. 2016zb009).

Acknowledgments: We would like to thank the researchers in the State Key Laboratory of Conservation and Utilization of Bio-Resources, Yunnan University (Kunming, China). Their very helpful comments and suggestions have led to an improved version of paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pandey, G.; Kumar, V.; Steinbach, M. *Computational Approaches for Gene Function Prediction: A Survey*; Department of Computer Science and Engineering, University of Minnesota: Minneapolis, MN, USA, 2006.
2. Altschul, S.F.; Gish, W.; Miller, W.; Myers, E.W.; Lipman, D.J. Basic local alignment search tool. *J. Mol. Biol.* **1990**, *215*, 403–410. [CrossRef]
3. Zacharaki, E.I. Prediction of gene function using a deep convolutional neural network ensemble. *PeerJ Comput. Sci.* **2017**, *3*, e124. [CrossRef]
4. Ofer, D.; Linial, M. ProFET: Feature engineering captures high-level protein functions. *Bioinformatics* **2015**, *31*, 3429–3436. [CrossRef] [PubMed]
5. Yu, G.; Rangwala, H.; Domeniconi, C.; Zhang, G.; Zhang, Z. Predicting gene function using multiple kernels. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2015**, *12*, 219–233. [PubMed]
6. Cao, R.; Cheng, J. Integrated protein function prediction by mining function associations, sequences, and protein-protein and gene-gene interaction networks. *Methods* **2016**, *93*, 84–91. [CrossRef] [PubMed]
7. Vascon, S.; Frasca, M.; Tripodi, R.; Valentini, G.; Pelillo, M. Protein Function Prediction as a Graph-Transduction Game. *Pattern Recogn. Lett.* **2018**. [CrossRef]
8. Radivojac, P.; Clark, W.T.; Oron, T.R.; Schnoes, A.M.; Wittkop, T.; Sokolov, A.; Graim, K.; Funk, C.; Verspoor, K.; Ben-Hur, A.; et al. A large-scale evaluation of computational gene function prediction. *Nat. Methods* **2013**, *10*, 221–227. [CrossRef] [PubMed]
9. Shehu, A.; Barbará, D.; Molloy, K. *A Survey of Computational Methods for Gene Function Prediction. Big Data Analytics in Genomics*; Springer International Publishing: New York, NY, USA, 2016; pp. 225–298.
10. Lobb, B.; Doxey, A.C. Novel function discovery through sequence and structural data mining. *Curr. Opin. Struct. Biol.* **2016**, *38*, 53–61. [CrossRef] [PubMed]
11. Njah, H.; Jamoussi, S.; Mahdi, W.; Elati, M. A Bayesian approach to construct Context-Specific Gene Ontology: Application to protein function prediction. In Proceedings of the 2016 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), Chiang Mai, Tailand, 5–7 October 2016.
12. Vens, C.; Struyf, J.; Schietgat, L.; Džeroski, S.; Blockeel, H. Decision trees for hierarchical multi-label classification. *Mach. Learn.* **2008**, *73*, 185–214. [CrossRef]
13. Liu, L.; Tang, L.; He, L.; Yao, S.; Zhou, W. Predicting gene function via multi-label supervised topic model on gene ontology. *Biotechnol. Biotechnol. Equip.* **2017**, *31*, 1–9. [CrossRef]
14. Ramage, D.; Hall, D.; Nallapati, R.; Nallapati, R.; Manning, C. LLDA: A supervised topic model for credit attribution in multi-Lcorpora. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP 2009, Singapore, 6–7 August 2009; pp. 248–256.
15. Mimno, D.; Mccallum, A. *Topic Models Conditioned on Arbitrary Features with Dirichlet-Multinomial Regression*; University of Massachusetts: Amherst, MA, USA, 2012; pp. 411–418.
16. La Rosa, M.; Fiannaca, A.; Rizzo, R.; Urso, A. Probabilistic topic modeling for the analysis and classification of genomic sequences. *BMC Bioinform.* **2015**, *16*, S2. [CrossRef] [PubMed]
17. Casella, G.; George, E.I. Explaining the Gibbs Sampler. *Am. Stat.* **1992**, *46*, 167–174.
18. Blei, D.M.; Kucukelbir, A.; Mcauliffe, J.D. Variational Inference: A Review for Statisticians. *J. Am. Stat. Assoc.* **2018**, *112*, 859–877. [CrossRef]
19. Tai, F.; Lin, H.T. Multilabel Classification with Principal Label Space Transformation. *Neural Comput.* **2012**, *24*, 2508. [CrossRef] [PubMed]
20. Sun, Y.; Ye, S.; Sun, Y.; Kameda, T. Improved algorithms for exact and approximate Boolean matrix decomposition. In Proceedings of the IEEE International Conference on Data Science and Advanced Analytics (DSAA), Paris, France, 19–21 October 2015; pp. 1–10.

21. Yang, Y. *Research on Biological Sequence Classification Based on Machine Learning Methods*; Shanghai Jiao Tong University: Shanghai, China, 2009.
22. Minling, Z.; Zhihua, Z. ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recogn.* **2007**, *40*, 2038–2048.
23. Zhang, M.L.; Zhou, Z.H. Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Trans. Knowl. Data Eng.* **2006**, *18*, 1338–1351. [CrossRef]
24. Tsoumakas, G.; Katakis, I.; Vlahavas, I. *Mining multi-label data. Data Mining and Knowledge Discovery Handbook*; Springer: New York, NY, USA, 2009; pp. 667–685.
25. Fan, R.E.; Chang, K.W.; Hsieh, C.J.; Wang, X.R.; Lin, C.J. LIBLINEAR: A Library for Large Linear Classification. *J. Mach. Learn. Res.* **2008**, *9*, 1871–1874.