Nigerian Journal of Technology (NIJOTECH)
Vol. 38, No. 1, January 2019, pp. **223 – 232**

Print ISSN: 0331-8443, Electronic ISSN: 2467-8821
www.nijotech.com
http://dx.doi.org/10.4314/njt.v38i1.27

EMPIRICAL PRIOR LATENT DIRICHLET ALLOCATION MODEL

M. A. Adegoke[1,*], J. O. A. Ayeni[2] and P. A. Adewole[3]
[1,] DEPARTMENT OF COMPUTER SCIENCE &TECHNOLOGY, BELLS UNIVERSITY OF TECHNOLOGY, OTA, OGUN STATE, NIGERIA
[2,] DEPT. OF COMPUTER SCIENCE, COLLEGE OF NATURAL SCIENCES, REDEEMER'S UNIVERSITY, EDE, OSUN STATE, NIGERIA
[3,] DEPARTMENT OF COMPUTER SCIENCES, FACULTY OF SCIENCE, UNIVERSITY OF LAGOS, AKOKA-YABA, LAGOS STATE, NIGERIA
E-mail addresses: [1] *adegoke_98@yahoo.co.uk,* [2] *ayenij@run.edu.ng,* [3] *padewolea@unilag.edu.ng*

ABSTRACT
In this study, empirical prior Dirichlet allocation (epLDA) model that uses latent semantic indexing framework to derive the priors required for topics computation from data is presented. The parameters of the priors so obtained are related to the parameters of the conventional LDA model using exponential function. The model was implemented and tested with benchmarked data and it achieves a prediction accuracy of 92.15%. It was observed that the epLDA model consistently outperforms the conventional LDA model on different datasets with an average percentage accuracy of 6.33%; this clearly demonstrates the advantage of using side information obtained from data for the computation of the mixture components.

Keywords: *latent Dirichlet allocation; semantic indexing; empirical prior; hidden structures; Prediction accuracy.*

1. INTRODUCTION

In modelling a collection of texts for information access tasks, the components of the textual collection called documents [1] can be represented as term vectors. This is described as a vector space model [2]. The vector space model (VSM) is a mathematical structure that organises the textual collection into a term-by-document which represents terms and corresponding counts for each document using matrix and vector notations. The vector space model is a mere observation of the terms; it does not reveal the latent structure or patterns that are present in the text collection [3]. One of the innovative and interesting approach for revealing the latent structure or patterns in a collection of textual items is a method called latent semantic indexing [4]. Latent semantic indexing (LSI) is an indexing procedure that addresses the deficiencies of the vector space model by using algebraically derived indices called single value decomposition to factor out the latent structure in a collection of textual documents. It has been shown that the algebraically derived vectors are more robust indicators of meaning than individual terms [5]. LSI approach has been successfully applied in document retrieval, text segmentation and text classification [6, 7]. However, its capabilities are limited when dealing with thematic content [8]. Though, LSI can well handle synonymous words but weak when handling polysemy [8 – 11]. Polysemous words are words that have multiple senses. LSI is thus effective when finding meaningful association among documents but does not provide an extension to deriving topics from the corresponding associative values. This is because the notion of topic is based on statistical properties of the corpus; however, LSI lacks the satisfactory and complete statistical foundation for topic derivation [12]. LSI model is therefore not strong enough to capture the intuition of topics in a document collection.

Probabilistic latent semantic analysis [8] is a probabilistic recast of LSI developed to address the weakness of LSI. Probabilistic latent semantic indexing (PLSI) suggests a probabilistic approach that captures the notion of topics in a collection of documents, as it is capable of handling both synonymous and polysemous terms [8, 13]. PLSI however has the limitation that the accuracy of its

***Corresponding author, tel: +234 – 705 – 892 – 7948**

results at the different run of the model varies. It has a local maximum due to the random initialisation of the underlying parameters [13, 11]. Also, PLSI model has no natural way of assigning probability to previously unseen documents [14]; this is because, it lacks ability to incorporate prior probability on the distribution over latent topics and distribution over words.

Latent Dirichlet allocation [13] is an extension of PLSI which, instead of randomly initialising model parameters, allocates fixed prior probabilities to the parameters. By considering a prior probability on these distributions, Latent Dirichlet allocation (LDA) model addresses the issue of the local maximum and defines a complete generative model. LDA is thus a probabilistic topic modelling algorithm that is capable of discovering hidden structures corresponding to themes and topics in a collection of textual data. In its original form, LDA has proven useful for modelling the latent structures and generating knowledge from corpora, which in many cases, were not possible with the previous text modelling approaches [15]. It has been applied to many types of problems including modelling scientific digital library [16], analysis of news articles [17], study of history of scientific ideas [18].

Though, LDA models have been widely used to identify hidden structures in data, the model suffers from the restriction that the value of its controlling parameters, namely, the prior beliefs for the computation of the hidden structures are not learnt or derived from data [1, 19]. Rather, fixed uniform priors are adopted and used irrespective of the nature and domain of application. There is no guarantee that the given priors are consistent with that of the underlying data. Learning the priors from data can improve model quality and greatly improve the quality of the inferred topics. In this paper, rather than using fixed uniform priors typical of the conventional LDA in its various modifications and extensions, we empirically construct the priors from data using latent semantic indexing algorithm. To validate the proposed model, namely, empirical prior latent Dirichlet allocation (*ep*LDA), we perform empirical study over a benchmark data.

2. RELATED WORKS

Latent Dirichlet allocation model explains the similarity of data by grouping features of these data into unobserved sets. Numerous flavours and reconfigurations have been developed around LDA. Modifications, extensions and improvements to the model are being developed and released at a rapid pace [1]. In order to maximise the likelihoods and ensure that knowledge from data is utilised in determining the optimal distribution of data, Blei and McAuliffe [20] introduced the supervised LDA (sLDA). In the sLDA, each document in a corpus is additionally associated with a value or word to indicate the group the document may be distributed. The algorithm takes into account the label on the constituent documents while maximising likelihoods. The label or the input value is to serve as the priors for the distribution parameters. This improvement is however not consistent with the original objective of the LDA model which is to generatively determine the best distribution of a text document. According to [21], annotating the features for a model that is supposed to identify unknown feature beforehand is hard to justify.

To allow the LDA model to handle multiple corpora during learning, Shen *et al* [22] developed collective latent Dirichlet allocation (C-LDA) which facilitates transfer of knowledge from one corpus to another. In order to facilitate efficient grouping of the features of related documents (data), Cheng and Blei [23] introduced a variant of LDA called relational topic model (RTM). RTM models documents and the link between them. Thus given a new document, RTM could be used to point the features that best describe the new document. However, the priors for the determination of the features are still prefixed, thus the quality of the inferred topics affects the effectiveness of the linking structure.

In order to improve the quality of the inferred topics by generating priors rather than allocating them, Wallach *et al* [24] introduced asymmetric-symmetric method. In the asymmetric- symmetric method, the commonly prefix prior over document-topic distribution, described as α, was obtained heuristically by varying the number of topics in the corpus until an optimal number of topics for that corpus is obtained. The topic number so obtained is later used to compute the prior while the commonly used prior over word proportion denoted by β is left intact. It, however, takes considerable experimentation to obtain an optimal number of topics and the experiment would have to be repeated every time a new corpus is to be analysed. A parallel alternative topic modelling algorithm to LDA is the non-negative matrix factorisation methods [25]. Non-negative matrix factorisation (NMF) methods are

however the same both in theory and results with PLSI in that they produce unstable results [25, 26]. Bayesian hierarchical kernelised probabilistic matrix factorisation algorithm [27] is a flavour of NMF that, instead of randomly generating the priors, attempts to incorporate row and column covariance structures as priors. Hierarchical algorithms are usually applied to spatial data where shape and density is often geometrically clear [28]. They become inept when dealing with documents since documents reside in very high dimensional space in which similarity is calculated using correlation instead of Euclidean distance. Thus, the hierarchical method works best as a matrix completion algorithm, and has no capacity to assign probability distribution to a test document.

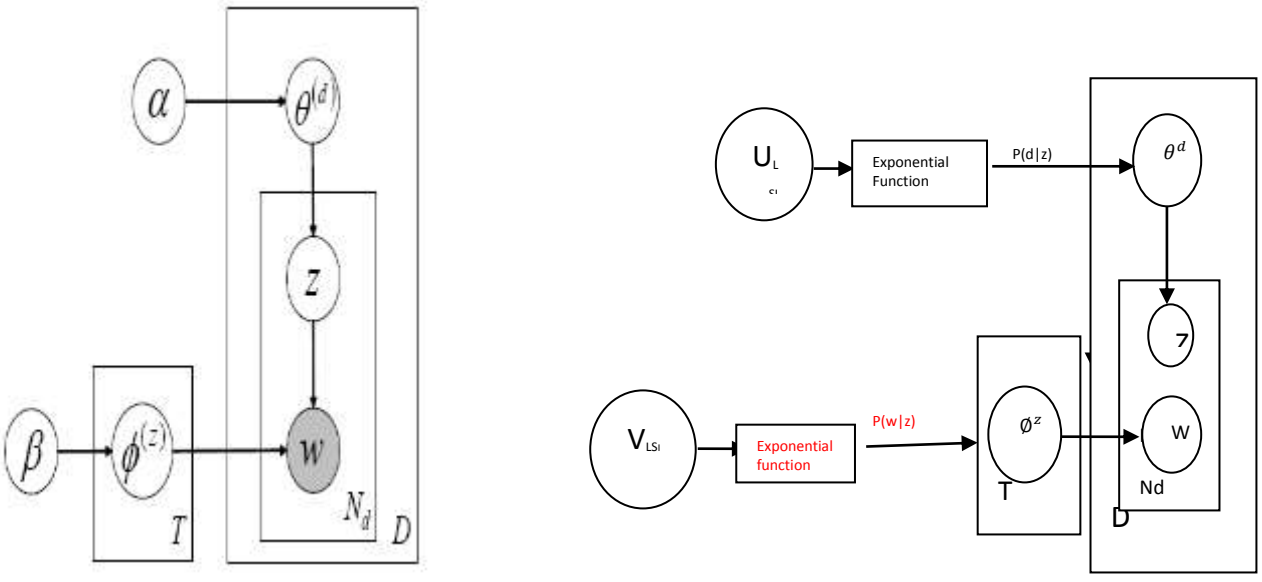
3. OVERVIEW OF LATENT DIRICHLET ALLOCATION

Given a collection of textual data, the underlying semantic structures that provide a complete description of the domain knowledge can be identified using latent Dirichlet allocation [13]. LDA is a class of topic modelling algorithms [8, 15] which describe a process that reveals the meaningful latent features corresponding to the themes or topics that are most prominent across a given corpus. The modelling process of LDA can be described as finding a mixture of topics z_i, $i = 1,...k$, for each document d. This mixture of topics is denoted by probability distribution $P(z_i|d)$ with each topic described by words, w_i which can be expressed as another probability distribution given by $P(w_i|z)$. Thus the set of words w_i that constitute each topic k is generated by first sampling a topic from the topic mixture $P(z_{i=k}|d)$ and then choosing a word from the probability distribution (of word over topic), $P(w_i|z_{i=k})$. This process can be expressed [29] as:

$$P(W_j) = \sum_{i=1}^{Z} P(w_i|z_{i=k})P(z_{i=k}|d) \quad (1)$$

where $P(w_i)$ is the probability of the ith word in a given document d and z_i is the hidden topic; $P(w_i|z_{i=k})$ is the probability of w_i within topic k, and $P(z_{i=k}|d)$ is the probability of picking a word from topic k in the document as stated earlier. The terms $P(w_i|z_{i=k})$ and $P(z_{i=k}|d)$ in equation (1) indicate which words are important for which topic and which topics are important for a particular document respectively.

Thus, the main objectives of LDA is to find the word distribution $P(w_i|z_{i=k})$ for each topic k and topic distribution $P(z_{i=k}|d)$ for each document d. Computing these distributions required that the prior distributions of the observed variables (words and documents) with the hidden variables $z_i$be known. Some methods initialise these parameters arbitrarily where the resulting model always reach local maximum [30]. LDA model however estimates these distributions using fixed priors, α and β. For notational convenience, let $\theta^d = P(z_{i=k}|d)$ and $\emptyset^z = P(w_i|z_{i=k})$. The generative process of LDA is illustrated in Figure 1 (a).



(a) (b)

Figure 1 – Graphical representation of (a) LDA and (b) epLDA.

where θ^d and $\emptyset^z$ are as defined above. α and β are the respective priors for θ^d and $\emptyset^z$. The inner plate N_d is the number of words for each document and the outer plate D is the total number of documents involved in the model.

Using Gibbs sampling procedure, the joint distribution of θ^d and $\emptyset^z$ can be used to compute the probability of assigning each word w_i in the text collection to each topic $z_i = k$ conditioned on prior topic assignments given by $z_{\neg i}$as follows [19, 14]:

$$P(z_{i=k}|z_{\neg_i}, wi) \propto (\theta^d|\alpha i)(\emptyset^z|\beta i) \quad (2)$$

Griffiths and Steyvers [31] showed that the estimation of each of the distributions θ^d and $\emptyset^z$ are obtained as follows:

$$\theta^d = \frac{n_d^z + \alpha}{\sum_k^Z (n_d^z + \alpha)} \quad (3)$$

$$\emptyset^z = \frac{n_z^w + \beta}{\sum_w^W (n_z^w + \beta)} \quad (4)$$

Thus equation (2) becomes:

$$P(z_{i=k}|z_{\neg_i}, wi) \propto \frac{n_z^w + \beta}{\sum_w^W (n_z^w + \beta)} \cdot \frac{n_d^z + \alpha}{\sum_k^Z (n_d^z + \alpha)} \quad (5)$$

where, n_d^z corresponds to likelihood of the association of document d to topic z, and α is the prior. Similarly, n_z^w corresponds to the likelihood of the association of word w to topic z, and β is the prior.

The values of α and β, often pre-fixed, cannot be generalised since they are not derived from the data [19, 1]. There is no guarantee that the given priors are consistent with the underlying data. It has been observed that inappropriate usage of priors through definitive allocation has resulted in some well developed models failing to produce reasonable predictions in real applications [27]. There is therefore a need for a technique that would seamlessly obtain the priors from the data so that the resulting model can be usable across different application domains. The thrust of this paper therefore is to adjust the original LDA such that the priors are obtained from sample data rather than by mere allocation. We nickname this model empirical prior LDA (*ep*LDA) and the graphical representation of the model is shown in Figure 1(b).

3.1 Empirical Prior Latent Dirichlet Allocation

An important property of the priors is that they are of the same exponential family with their respective multinomial distributions [19]. That is, the priors α and β are of the form *P(z|d)* and *P(w|z)* respectively. Thus any model that is capable of generating *P(z|d)* and *P(w|z)* from a sample text data will be a good candidate for deriving the priors from data. Latent Semantic Indexing (LSI) model [4], using singular value decomposition (SVD), an algebraic method, has been reported of capable of generating the algebraic equivalence of these parameters [8] as U and V respectively; where U and V are matrices define by: U = (*d* x *z*) and V = (*w* x *z*). Matrices U and V attempts to associate documents d_i and word w_i respectively to the underlying themes z_i in each of the decomposed matrices. As stated earlier, LSI cannot be used to obtain topics directly [12]; It can however be used to obtain a rough associate of words and documents in a textual collection to the underlying themes. This property of LSI is therefore exploited to obtain the priors from data. The transpose U^t of matrix U is obtained to reflect the needed *z* x *d* matrix. Note that, the elements of LSI matrices U and V are numeric and are not probability values therefore, they are not directly interpretable to *P(z|d)* and *P(w|z).* However, under certain assumptions, probability model can be defined for the LSI factors U and V.

The probability interpretation is defined using the hypothesis that a person writing a document has certain themes in mind. Consequently, documents are not arranged haphazardly but according to certain underlying themes (latent structures). Thus, the distributions of documents in a corpus do not occur randomly but according to these latent structures [32]. Specifically, assuming a document d_i is characterised by a hidden structure c, the distribution of the document (observed variables) can be related to its hidden structure by the following [33] exponential functions:

$$P(di\ |c) = \frac{e^{(\boldsymbol{d_i}.\boldsymbol{c})^2}}{Z(\boldsymbol{c})} \quad (6)$$

For documents with k characteristics hidden structures, $c_1 \ldots c_k$, the probability interpretation of the elements of the underlying structures $c_1 \ldots c_k$ in a corpus D = [d_1, …,d_n] can be generalised as:

$$P(D|c_1 \ldots c_k) = \frac{e^{(d_i.c_1)^2 + \cdots + (d_i.c_k)^2}}{Z(\boldsymbol{c_1} \ldots \boldsymbol{c_k})} \quad (7)$$

where, $c_1 \ldots c_k$ are the elements of the association of the documents to the hidden themes and $Z(c_1 \ldots c_k)$ is the normalisation factor to ensure that each row adds up to 1.

Now using LSI to obtain the hidden structures where the elements of the matrix U = *d x z* that associates documents d_i to its respective hidden structures is given by $u_1 \ldots u_k$, the probability interpretation of the

latent structures is defined according to the following distribution:

$$P(D|u_1 \dots u_k) = \frac{e^{(d_i.u_1)^2+\cdots+(d_i.u_k)^2}}{Z(u_1 \dots u_k)} \quad (8)$$

Thus the probability interpretation of each elements of the LSI matrix U that associates document d_i to its hidden structure z_i is obtained by:

$$P(di|zi) = \frac{e^{(d_i.u_{ij})^2}}{z(u_j)} \quad (9)$$

where $u_j = [u_{i,j}]$, $j = 1,\dots,k$ are the elements of row u_j and each $z(u_j)$ is an integral in the row space to normalise the respective row. The integral changes to a sum $\sum_{i=1}^{d} e^{(d_i.u_{ij})^2}$ for discrete variables such as documents.

This brings equation (9) to:

$$P(di|zj) = \frac{e^{(d_i.u_{ij})^2}}{\sum_{i=1}^{d} e^{(x_i.u_{ij})^2}} \quad (10)$$

Equation (10) relates each element of matrix U of LSI model to its probability equivalence. Thus instead of assuming that all documents have the same chance of being allocated to a topic z_j, the respective likelihood are obtained from data using the LSI framework. The prior association of $n_{d_i}^{z_j}$ which used to be α as we have in equation (3) has now become $P(d_i|z_j)$ which is obtained from data as expressed in equation (10).

Similarly, the discussion holds for V in which the probability that term w is associated to a topic zis represented by V as follows:

$$P(w_i|v_i \dots v_k) = \frac{e^{(w_i.v_1)^2+\cdots+(w_i.v_k)^2}}{Z(V_1 \dots . V_k)} \quad (11)$$

where $Z(v_1 \dots . v_k)$ is normalisation factor with each $z(v) = \sum_{w=1}^{V} e^{(w_i.v_{ki})^2}$ thus, the probability interpretation of each $v_{i,j}$ is given by:

$$P(wi|zi) = \frac{e^{(w_i.v_{ki})^2}}{\sum_{t=1}^{d} e^{(t_i.v_{ki})^2}} \quad (12)$$

Instead of assuming that all terms have the same chance of being associated to the hidden topic z, rather, the probability depends on the prior associations. The prior association of the likelihood $n_{d_j}^{w_i}$ is given by $P(w_i|z_j)$. Bringing these non-uniform priors to replace the fixed assumptions of α and β in expression (5) gives expression (13) as shown at the bottom of this page.

This can be expanded by substituting $P(w_i|z_j)$ *and* $P(d_i|z_j)$ as we have in equations (10) and (12) to yield the following expression (14) also at the bottom of the page.

Expression (14) embeds LSI factorization for the purpose of capturing the prior distributions from data instead of using fixed uniform priors for these important parameters. This constitutes the empirical prior latent Dirichlet allocation (*ep*LDA) model. The model can be used across different application domains because the priors are not fixed but are evaluated from the data. We state the procedure for obtaining the set of topics in the empirical prior Latent Dirichlet Allocation Model. This is a prelude to classifying the original documents and thereafter using the results for prediction task. As a module, *ep*LDA can be embedded in a more complex model for classification and prediction tasks. The overall procedure for the empirical prior Dirichlet allocation is given as follows:

Input: Collection of textual documents

Output: Term-by-document matrix X = *t x d*, LSI matrices U = *z x d* and V = *w x z*, probability interpretation of U and V given by $p(z|d)$ and $p(w|z)$, set of topics obtained from corpus by $p(z_i = j|z_{\neg i}, w, d)$.

Process:

Step 1: Remove stop words to form the vocabulary list.

Step 2: Generate term-document matrix from the vocabulary list, normalise to obtain matrix X.

$$\begin{array}{c} X \\ (\mathbf{d}_j) \\ \downarrow \\ (\mathbf{t}_i^T) \rightarrow \end{array} \begin{bmatrix} x_{1,1} & \cdots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \cdots & x_{m,n} \end{bmatrix}$$

$$\mathrm{P}(z_i = \mathrm{k}|z_{\neg i}, \mathrm{wi}) \propto \frac{n_{d_j}^{w_i} + P(w_i|z_j)}{\sum_k^Z (n_{d_j}^{w_i} + P(w_i|z_j))} \cdot \frac{n_{z_j}^{w_i} + P(d_i|z_j)^t}{\sum_w^W (n_{z_j}^{w_i} + P(d_i|z_j)^t)} \quad (13)$$

$$\mathrm{P}(z_i = \mathrm{k}|z_{\neg i}, \mathrm{wi}) \propto \frac{n_{z_j}^{w_i} + \frac{e^{(w_i,z_j)^2}}{\sum_{k=1}^{W}(w_k,z_j)^2}}{\sum_w^W (n_{z_j}^{w_i} + \frac{e^{(w_i,z_j)^2}}{\sum_{k=1}^{W}(w_k,z_j)^2})} \cdot \frac{n_{d_j}^{w_i} + \frac{e^{(z_j,d_i)^2}}{\sum_{k=1}^{D}(z_j,d_k)^2}}{\sum_k^Z (n_{d_j}^{w_i} + \frac{e^{(z_j,d_i)^2}}{\sum_{k=1}^{D}(z_j,d_k)^2})} \quad (14)$$

Step 3:Decompose the term-document matrix into three (3) orthogonal matrices U^T, Σ, V using SVD algorithm available in matrix toolkit:

$$\underset{(\mathbf{t}_i^T)\rightarrow}{\overset{X}{\underset{(\mathbf{d}_j)\downarrow}{}}} \begin{bmatrix} x_{1,1} & \cdots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \cdots & x_{m,n} \end{bmatrix} = \underset{(\hat{\mathbf{t}}_i^T)\rightarrow}{U^T} \begin{bmatrix} \begin{bmatrix} \mathbf{u}_1 \end{bmatrix} & \cdots & \begin{bmatrix} \mathbf{u}_l \end{bmatrix} \end{bmatrix} \cdot \underset{}{\Sigma} \begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_l \end{bmatrix} \cdot \underset{(\hat{\mathbf{d}}_j)\downarrow}{V} \begin{bmatrix} [& \mathbf{v}_1 &] \\ & \vdots & \\ [& \mathbf{v}_l &] \end{bmatrix}$$

Step 4: Obtain the probability equivalence of the LSI factors U and V obtained in step3 using equations (10) and (12) to obtain the parameters,α= *P(d/z)* and β = *P(w/z)*

where: d = $\{x_i\}$i = 1,…,k and w = $\{t_i\}$i = 1,…,k

Step 5: Use the modified expression (14) to generate the relevant data (topics) in a collection of textual documents.

Step 6: end.

4. EXPERIMENTAL RESULTS

In this section, we evaluate the performance of the proposed *ep*LDA model on a benchmarked data. In particular, the performance of the proposed model is evaluated on the real roll call data of the United States Congress to perform legislative prediction. The roll call data was chosen for the evaluation because the same data have been used in previous studies.

4.1 Data set

Roll call data, otherwise called legislative history, containing voting results *yea* or *nay*, and content of the legislative bills are available in Govtrack [34], an independent website which provides comprehensive information to the public. The real roll call data for six (6) years spanning three (3) congregational sessions containing bills from 109th through 111th congregational session was used in this implementation.

Gerrish and Blei [35] used real data from 106th to 111th congregational sessions, but focused their analyses only on the 111th session; while, Yang and Wang [27] used data from 111th congregational session only. We therefore implemented the proposed model on real roll call data for 111th congregational session so as to compare our results with [35, 36, 27] that used sLAD, term-document frequency, and Bayesian Hierarchical Kernelised Probabilistic Matrix Factorisation (BH-KPMF) models respectively to model the text bills on the same data set for a predictive task. We later extended our set of data backward to include roll call data for 110th and 109th congregational sessions.

4.2 Experimental Settings

The real roll call data for each of 111th, 110th and 109th congregational sessions were randomly partitioned into two sets of training and test in the ratio of 80% to 20% respectively. Ten (10) random samples of training and test sets are obtain from each dataset. The training set was used to learn the model, while the test set is treated as previously unseen data (new bills whose votes is to be predicted). Following [27, 35, 36] that used votes from 80 active legislators over 120 bills from the 111th congregational session, 120 bills from each congregational session were used. The proposed *ep*LDA was then used to model the training set into the underlying topics, which provide insight into what drives the voting pattern. It was observed from the simulation results that topics converge (i.e., no new topics were formed after the number of topics exceeded 20. This study therefore sets the number of topics for each dataset to 20. This is consistent with the settings in previous works [35].

For comparison study, the proposed *ep*LDA was compared with sLDA [35], RWHG [37], Influence Network [38], and BH-KPMF [27], all of which used the same datasets for prediction. Since topic models require the number of topics to be set before learning begins, different values of topic numbers were tested. From the simulation results it was observed that no more topics were formed after the number of topics exceeded 20. We therefore heuristically set the numbers of topics for each congregational session to 20. This is consistent with the setting in sLDA, a topic modelling technique.

4.3 Results

Recall that the goal of topic modelling algorithms is to discover the latent topics that are embedded in a textual collection; and a common way to display these topics is to index the highest probability words for each topic [1, 15], where a topic is described by vector of words that best describes a theme. Figure 2 shows the results for the 20 topics in the text bills of the 111th congregational session.

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
certain	motion	number	amdt	act
unit	ensign	medicar	entiti	relat
state	from	save	health	provid
limit	senat	improve	judg	nomin
prevent	elimin	nation	program	reid
modify	plan	response	or	nature
at	benefit	class	proceed	u
service	reduce	this	district	confirm
ensur	famili	new	secure	passage
amount	caregiv	social	with	financ
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
Topic 16	Topic 17	Topic 18	Topic 19	Topic 20
no	h	r	s	of
substitute	tabl	invoke	make	amend
care	patient	afford	report	bill
veteran	military	fund	protect	appropri
hous	re	coburn	committee	purpose
circuit	concur	commit	defence	fiscal
avail	recommit	develop	individual	cloture
use	individual	waiv	hold	depart
rule	hold	guantanamo	temporarili	septemb
science	suspend	held	perman	year

Figure 2: Topics of the Bills for the 111th Congregational Session

Each set of the derived vectors serves as a topic/category to which a test bill could be classified. These topics describe the underlying patterns in the textual collection (legislative bills) to which the texts could be classified. A text bill is classified to a topic if they (the topic and the bill) conform to the same parametric distribution [28]. Both the training bills and test bills are therefore classified into topics. Once the bills are grouped this way, the chance or probability of legislator x_i voting *yea* for a new bill is based on the votes of previous bills to which the new bill is classified. This is based on the beliefs of the United States Congress that voting pattern is determined by the information embedded in the bill rather than by party affiliation [37]. Therefore, bills with similar information tend to receive similar votes from the same set of legislators. Thus, votes for each bill are predicted by votes of the category to which it is classified. That is, votes on an unlegislated bill are estimated based on the prior votes for similar bills from the same legislator. Specifically, the votes of each legislator *x* on a new bill *y(k+1)* is estimated *yea* or *nay* by the following probability function [37, 35]:

$$P\left(Xi = yea \middle| Y(k+1)\right) = \frac{P_{(y_k)}^{yea}}{P_{(y_k)}^{yea} + P_{(y_k)}^{nay}} \quad ; k = 1,2,\dots,n \tag{15}$$

Where, $P(X_i = yea|Y_{(k+1)}) = \begin{cases} yea, & if \ \frac{P_{(y_k)}^{yea}}{P_{(y_k)}^{yea}+P_{(y_k)}^{nay}} \geq 0.5 \\ nay, otherwise. \end{cases}$

Thus, the ability to effectively group the bills correctly is an important factor in determining the voting pattern on a new bill [35, 37].

The performance evaluation of topic models is usually measured using prediction accuracy metrics [19] given by the proportion of votes correctly predicted over the total number of votes. That is,

%Prediction accuracy =

$$\frac{No.of\ votes\ predicted\ correctly}{Total\ No\ of\ votes} \times 100 \tag{16}$$

Table 1 shows the average percentage accuracy for each of the congregational sessions using the proposed *ep*LDA and the conventional LDA models. While Table 2 compares the average percentage prediction accuracies of the *ep*LDA with other similar models in literature on the same dataset. The comparison is graphically display in Figure 3.

Table 1 – Average % Prediction Accuracy for each of the Congregational Sessions Using epLDA and LDA Based Classifiers

Model Performance Metric/congregational sessions	111th Congregational Session	110th Congregational Session	109th Congregational Session
*ep*LDA	92.12	92.48	91.73
LDA	85.1	81.25	91.0

Table 2 – Comparison with the results of other models on 111th congregational dataset.

Model Description	% Accuracy	% Error rate
sLDA by (Gerrish and Blei,2011)	87.0%	13.0%
RWHG by (Wang et al., 2011)	90.36%	9.6%
Sampling Approach Influence Network (Hanneke, 2010)	81.3%	18.75%
BH-KPMF (Yang & Wang, 2014)	90.14%	9.86%
LDA model Based	85.1%	14.9%
*ep*LDA model Based	92.15%	7.5%

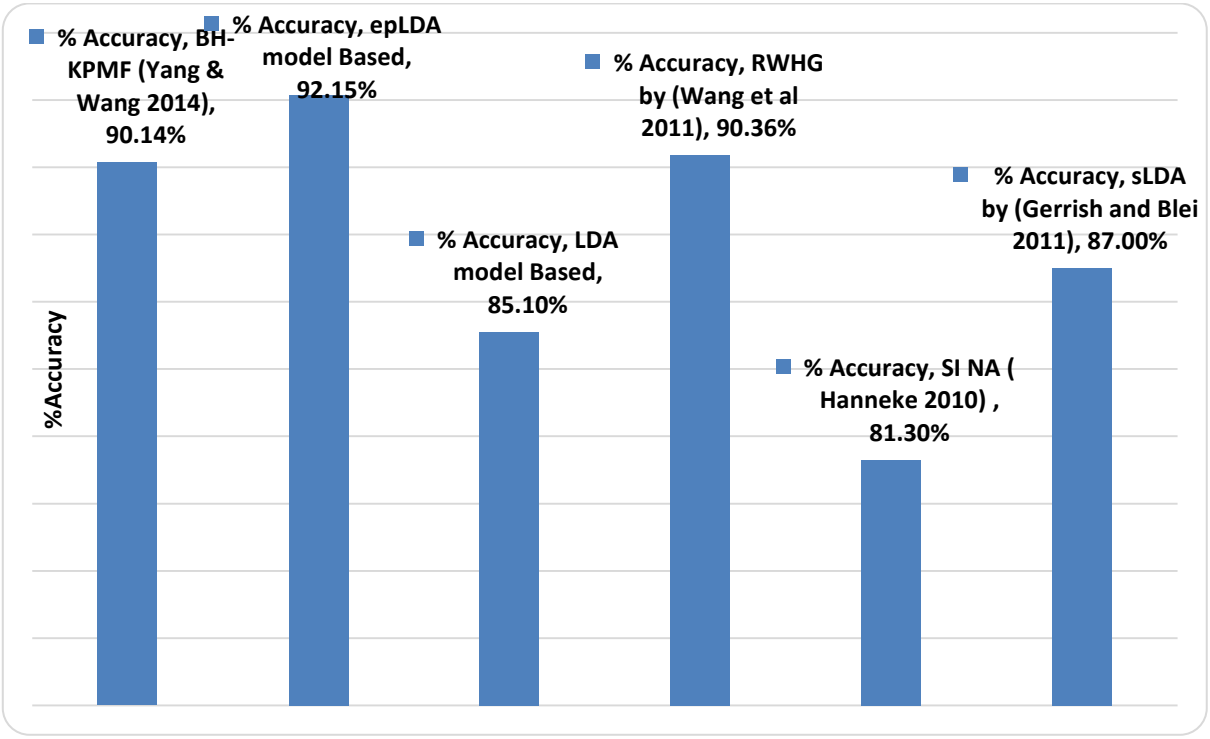


Figure 3 - Comparing Results with Related Models

From the results in Table 1 it is easy to see that the prediction accuracy is consistent across the different dataset for the *ep*LDA model. That is, irrespective of the set of documents, the % accuracy is consistent for the *ep*LDA model based classifier. On the other hand, there is a remarkable difference in the average % prediction accuracies for the three different dataset using LDA model based classifier. This clearly demonstrates the advantage of incorporating side information derived from data for the modelling of the text data for prediction task instead of just assuming fixed uniform prior information, which may not be consistent with the underlying data across the different datasets. Thus, the proposed *ep*LDA leverages the side information obtained from data for the predictive ability, resulting in significant performance gain.

4.4 CONCLUSION

In this work, an extension to LDA model namely, empirical prior latent dirichlet allocation (*ep*LDA) model was formulated, implemented and tested with real data. The proposed topic model has the capacity to obtain prior knowledge needed for the computation of the hidden structures of a collection of data items from the data itself. The key idea of themodel is the incorporation of some flexibility to the original LDA model with the aim of enhancing its generalisation and performance. Rather than pre-allocating fixed prior values for the computation of the hidden structures in a collection of discrete data irrespective of the domain of application, the proposed *ep*LDA obtains this prior knowledge from the data to be processed. This enables the model to be usable across different application domains since the model is able to dynamically pick the prior

information from the data. Experimental results on real datasets demonstrate the effectiveness of the proposed model. Compared with sLDA and BH-KPMF and other pattern recognition models as seen in Figure 3, *ep*LDA produces superior performance. As future work, we wish to demonstrate the practical application of the empirical prior LDA (*ep*LDA) in the domain of software requirements analysis.

5. REFERENCES

[1] Gross A. & Murthy D. Modelling virtual organisation with Latent DirichletAllocation: A case for natural language processing. *Neural Networks* 58 pp38-49. 2014

[2] Salton G.. A Theory of term importance in automatic text indexing. *Communications of the ACM,* 18(11), pp613-620. 1975

[3] Dubin D. The most influential paper Gerald Salton never wrote. *Library Trends,* vol. 52, No. 4, pp748-764. Illnois. 2004.

[4] Deerwester S., Dumais S.T., Frnas G. W., Landauer T. R.. Indexing by Latent Semantic Analysis. *Journal of the American Society of Information Science*. 41 (6): pp391-407, 1990.

[5] Berry M.W., Susan T.D., & Gavin W.O.,. Using linear algebra for intelligent information retrieval. *SIAM Review*, 37: 4, pp.573-595, 1995.

[6] Brants T., Chen F., & Tsochantaridis. Topic-based document segmentation with probabilistic latent semantic analysis. *In proceedings of conference on Information & knowledge management*, pp 211-218, 2002.

[7] Wu H &Gunopulos D. Evaluating the utility of statistical phrases and latent semantic indexing for text classification. *In the proceedings of IEEE International Conference on Data Mining,* PP713-716, 2002

[8] Hoffmann T. Probabilistic latent semantic indexing. *In proceedings of SIGIR' 99*, pp 35-44. 1999,

[9] Hofmann, T.. Unsupervised Learning by Probabilistic Latent Semantic Analysis. *Machine Learning Journal, 42(1)*, 177-196. 2001.

[10] Martin D. I., and Berry M.W. Handbook of latent semantic analysis. Taylor and Francis group, New York: **name of Publisher missing**, 2007.

[11] Farahat A. O. and F. R. Chen. Improving probabilistic latent semantic analysis using Principal component analysis. *In Eleventh Conference of the European Chapter of the Association for Computational Linguistics* (EACL -2006). 2006.

[12] Kathry B. Laskey and Henri Prade, eds. *UAI '99: Proceedings of the Fifteenth Conference on uncertainty in Artificial intelligence, Stockholm, Sweden, July 30 – august 1, 1999.* Morgan Kaufmann.

[13] Blei D.M., Ng A.Y., and Jordan M.I. Latent dirichlet Allocation. *Journal of Machine Learning Research* 3: pp. 993-1022. 2003.

[14] Landuer T.K., McNarama D.S., Dennis S., & Kintsch W.. *Handbook of latent semantic analysis.* Lawrence Erlbaum, Mahwah, NJ. 2007.

[15] Blei, D. M.. *Probabilistic topic models. Communications of the ACM,* 55(4), 77–84. 2012.

[16] Mann, G. S.,&Mimno, D. et al.. Bibliometric impact measures leveraging topic analysis. *In Proceedings of the 6th ACM/IEEE-CS joint conference on digital libraries*. JCDL'06, 2006.

[17] Newman, D., Chemudugunta, C., Smyth, P. Statistical entity-topic models. In: *SIGKDD, ACM* , pp. 680-686, 2006.

[18] Blei, D., and Lafferty, J.. Topic models. In Srivastava, A., and Sahami, M., eds., *Text Mining: Theory and Applications.* Taylor and Francis. 2009.

[19] Heinrich George. Parameter estimation for text analysis. Technical report, University of Leipzig, Germany, version 2.5, 2009,

[20] Blei, D. M., & McAuliffe, J. D. Supervised topic models. *ArXiv PreprintarXiv*:1003.0783. 2010.

[21] Williamson S., Orbanz P. & Ghahramanani.. Dependent Indian Buffet process. In the proceedings of the *13th International Conference on artificial intelligence and statistics (AISTATS).* Chia Laguma Resort, Sardina, Italy. Vol 9 of JMLR. 2010.

[22] Shen Z. Y., Sun J., & Shen Y.D.,. Collective Latent Dirichlet Allocation. *Eighth IEEE International Conference on Data Mining*. Pp. 1019-1024. 2008.

[23] Cheng J. &Blei D. M. Realtional Topic Models for Documents Network. *12th International conference of Artificial Intelligence & Statistics* (AISTATS); Vol 5; Florida, USA. 2009.

[24] Wallac H. M., Mimno D., McCallum A.,. Rethink LDA: Why priors Matter. http://rexa.info, 2010.

[25] GreeneNom-negative matrix factorisation for topic modelling. Derek Greene's Home. Machine learning. http://derekgreene.com/ nmf-topic. 2015.

[26] Andrew P. Non-negative matrix factorisation and probabilistic latent semantic analysis. 2011.

[27] Yang H., and Wang J., Bayesian Hierarchical Kernelized Probabilistic Matrix factorization. Communications in Statistics – *Simulation and Computation,* 2014.

[28] Duan, C. Clustering and its Application in Requirements Engineering, Technical Report #08-001, School of Computing., DePaul University, Available online at http://www.cs.depaul.edu. 2008.

[29] Krestel R., Fankhauser P., and Nejdl W. Latent dirichlet allocation for Tag Recommendation. *In RecSys'09. Association of Computing Machinery*. New York, USA. ACM 978-1-60558-435-5/09/10. 2009.

[30] Russel S and Norvig P. *Artificial intelligence, A Modern Approach* 2nd Edition. Pearson Education, Prentice Hall. Upper saddle River NJ 07458. www.prehall.com, 2011.

[31] Griffitths T. L., and Steyvers M. Finding Scientific topics. *Proceedings of the National Academy of Sciences*, 101(1): 5228-5235. 2004.

[32] Tipping M. and Bishop C. Principal component analysis. *Journal of the Royal Statistics society, series B,* 61(3); p. 611-622. 1999.

[33] Ding H.Q.. A Similarity-based Probabilistic Model for Latent Semantic Indexing. *SIGIR '99.* Berkely, CA, USA, pp58-65. 1999.

[34] Tauberer J. Govtrack. (http://www.govtrack.us/data/us/,2012). 2012.

35] Gerrish S. M. and Blei D. M. Predicting legislative roll calls from text, *Proc. Int. Conf. Mach. Learn.* (Bellevue, WA), Jun.–Jul. 2011, pp. 489–496. 2011.

[36] Goldblatt D. & O'Neil T. How Bill Becomes a Law – Predicting Votes from Legislation Text. Joint project of CS 224n and CS 229, Stanford University.*cs229.stanford.edu/proj2012/GoldblattONeil.* 2012.

[37] Wang J., Varshney K. R., and Mojsilovi´c A.. Legislative prediction via random walks over a Heterogeneous graph. In *SIAM International Conference on Data Mining*, 2011. 2011.

[38] Hanneke S.. A structural approach to legislative roll call vote prediction. http://www.cs.edu/~sha,2010.