The Deep Latent Position Topic Model for Clustering and Representation of Networks with Textual Edges

Rémi Boutin, Pierre Latouche, Charles Bouveyron, Dingge Liang

The Deep Latent Position Topic Model for Clustering and Representation of Networks with Textual Edges

Rémi Boutin [1] & Pierre Latouche [1,2] & Charles Bouveyron[3] & Dingge Liang[3]

[1] *Université Paris Cité, CNRS, Laboratoire MAP5, UMR 8245, Paris, France*
[2] *Université Clermont Auvergne, CNRS, LMBP UMR 6620, Aubière, France*
[3] *Université Côte d'Azur; INRIA, CNRS, Laboratoire J.A.Dieudonné, Maasai team, Nice, France*

Résumé. De nombreuses interactions numériques amènent les utilisateurs à partager du contenu publié par d'autres. Ce type de données est naturellement représenté par un réseau ayant pour nœuds les comptes des utilisateurs et pour arêtes, les documents partagés. Pour comprendre ces structures complexes et hétérogènes, il est crucial de pouvoir classifier les nœuds en groupes homogènes et d'obtenir une visualisation du réseau interprétable. Pour répondre à ces deux problématiques, nous présentons Deep-LPTM, un modèle de classification de nœuds reposant sur un auto-encodeur de graph variationnel ainsi que sur un modèle probabiliste profond pour la représentation simultanées des documents et des nœuds dans deux espaces latents. Les paramètres sont estimés à l'aide d'un algorithme d'inférence variationnelle. Le modèle est évalué sur des données synthétiques et est comparé avec l'état de l'art, à savoir ETSBM et STBM.

Mots-clés. Réseaux de neurones convolutif pour les graphes, Modèle de thèmes plongés, Modèle à positions latentes profonds, Apprentissage non-supervisé

Abstract. Numerical interactions leading to users sharing content published by others are naturally represented by a network where the individuals are associated with the nodes and the exchanged texts with the edges. To understand those heterogeneous and complex data structures, clustering nodes into homogeneous groups is crucial as well as rendering an comprehensible visualisation of the data. To address both issues, we introduce Deep-LPTM, a model-based clustering strategy relying on a variational graph autoencoder approach as well as a probabilistic model to characterise the topics of discussion. Deep-LPTM allows to build a joint representation of the nodes and of the edges in two embeddings spaces. The parameters are inferred using a variational inference algorithm. An extensive benchmark study on synthetic data is provided. In particular, we find that Deep-LPTM better recovers the partitions of the nodes than the state-of-the art ETSBM and STBM.

Keywords. graph convolutional network, embedded topic model, deep latent position model, unsupervised learning

1 Introduction and notations

Numerical interactions between individuals often imply the creation of texts. For instance, on social media such as Twitter, it is possible to publish some content, a tweet or a post, that will in turn be republished, or re-twitted, by other accounts. Also, it is possible to mention another account directly in the publication. In the same way, the exchange of mails between collaborators can be seen as connections between accounts exchanging documents. Both examples can be represented by a network with the nodes corresponding to the accounts, and the edges to the exchanged texts. This complex data structure is difficult for the human to apprehend, due to the heterogeneity of the data, and in particular when considering massive data. One solution is to cluster homogeneous nodes into groups to obtain intelligible and useful information. However, very few methods performing node clustering are able to simultaneously exploit both the texts present on the edges and the connections. Moreover, they do not provide direct means to represent the network as illustrated in Figure 1. Consequently, we introduce Deep-LPTM whose generative assumptions are presented in Section 2. Section 3 focuses on the inference and Section 4 provides the evaluation of the model against state-of-the-art methods. In section 5, we briefly discuss extension of this work that will be presented if the paper is accepted.

In this paper, we are interested in data represented by a graph $\mathcal{G} := \{\mathcal{V}, \mathcal{E}\}$ where $\mathcal{V} = \{1, \dots, N\}$ denotes the set of vertices. The set $\mathcal{E}$ denotes the edges between the nodes with $M = |\mathcal{E}|$ the number of edges. We focus on binary a adjacency matrix $A \in \mathcal{M}_{N\times N}(\{0, 1\})$ such that A_{ij} equals 1 if $(i, j) \in \mathcal{E}$, and 0 otherwise. The graph is assumed to be directed and without any self loop. Therefore $A_{ii} = 0$ for all $i \in \mathcal{V}$. Finally, Q denotes the number of clusters of nodes.

Each edge in the graph represents a textual document sent from one node to another. An edge from node i to node j exists or equivalently $(i, j) \in \mathcal{E}$, if and only if node i sent a textual document to node j, denoted W_{ij}. We use a bag-of-word representation of the texts where $W_{ij} = \left(W_{ij}^1, \dots, W_{ij}^V\right) \in \mathbb{N}^V$ denotes the vector of word occurrences in the document between nodes i and j such that W_{ij}^v is the number of times word v appears in the document, $M_{ij} = \sum_{v=1}^{V} W_{ij}^v$ is the total number of words in document W_{ij} and V the size of the vocabulary. The set of documents will be denoted $W := (W_{ij})_{(i,j)\in\mathcal{E}}$ and the number of topics is denoted by K. Eventually, the simplex of dimension d will be denoted Δ_{d-1}.

2 Model

In the following, the assumptions about the graph generation as well as the hypothesis concerning the documents construction are presented.

Graph generation Assuming that the number of clusters Q is fixed before hand, each node i is assumed to belong to a cluster, represented by the cluster membership variable C_i. The variables C_i, for any $i \in \mathcal{V}$, are assumed to be independent and identically distributed

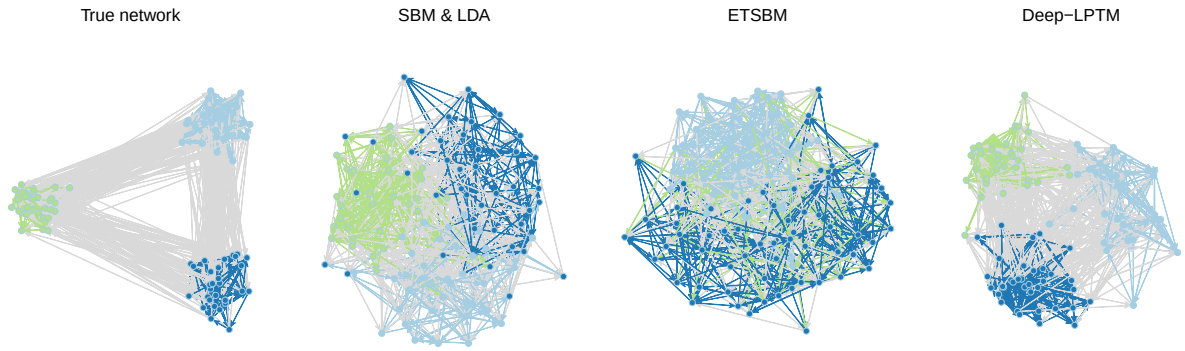

Figure 1: Illustration of Deep-LPTM contributions. The node (edge respectively) colours denote the corresponding clusters (topics). The true partitions of the network are represented on the left hand side. The second figure relies on the node clusters and the topics rendered by the stochastic block model (SBM) and the latent Dirichlet allocation (LDA) respectively. The third network is based on the state-of-the-art embedded topics for the stochastic block model (ETSBM). Finally, the last network renders Deep-LPTM node clustering and latent node positions. The former methods do not provide node positions.

(i.i.d) according to a multinomial distribution such that for any node $i \in \{1, \dots, N\}$:

$$C_i \sim \mathcal{M}_Q(1, \pi), \tag{1}$$

with $\pi \in \Delta_{Q-1}$ and $C_i \in \{0,1\}^Q$ being one hot encoded so that $C_{iq} = 1$ if node i belongs to cluster q and $C_{iq} = 0$ otherwise. Thus, denoting $C = (C_1, \dots, C_N)^T \in \mathcal{M}_{N\times Q}(\{0,1\})$ the cluster membership matrix, we have:

$$p(C) = \prod_{i=1}^{N} \prod_{q=1}^{Q} \pi_q^{C_{iq}}. \tag{2}$$

Moreover, given its cluster membership, the node i is assumed to be represented by a Gaussian vector Z_i in a p dimensional latent space such that:

$$Z_i \mid C_{iq} = 1 \sim \mathcal{N}\left(\mu_q, \sigma_q^2 I_p\right). \tag{3}$$

Eventually, the connection between two nodes is assumed to depend on the closeness of the node representations in the latent space. Therefore, denoting $\eta_{ij} := \kappa - \|Z_i - Z_j\|$, the probability for node i to be connected to node j is:

$$P(A_{ij} = 1 \mid Z_i, Z_j) = \frac{1}{1 + e^{-\eta_{ij}}}, \tag{4}$$

where a logistic function is used as a link function. For the sake of brevity, we will denote $p_{ij} = (1 + e^{-\eta_{ij}})^{-1}$.

It is worth noticing that the model described in Equations (1), (3) and (4) corresponds to the latent position cluster model Handcock et al. (2007). The fundamental difference with our approach for this part of the model will arise in the inference, as discussed in Section 3.

Generation of the texts on the edges At the core of our approach is the motivation to use textual data to obtain more homogeneous and meaningful clusters. To begin with, we make the assumption that each edge can be represented in a latent space by a Gaussian vector, depending only on the node cluster memberships. Thus, given $(C_i)_{i\in\mathcal{V}}$, the latent variables Y_{ij} are assumed to be i.i.d such that:

$$Y_{ij} \mid A_{ij}C_{iq}C_{jr} = 1 \sim \mathcal{N}(m_{qr}, \text{diag}(s_{qr}^2)),\ \forall (i,j) \in \mathcal{E}, \tag{5}$$

where $m_{qr} \in \mathbb{R}^K, s_{qr} \in \mathbb{R}^K$. Moreover, we assume that the topic proportions of the document W_{ij}, denoted θ_{ij}, can be deduced from the latent variables such that: $\theta_{ij} = \text{softmax}(Y_{ij})$ where $\text{softmax}(x) = \left(\sum_{k=1}^K e_k^x\right)^{-1} (e^{x_1}, \ldots, e^{x_K})^\top$. Hence, assuming that the documents are i.i.d given their corresponding topic proportions, we have for any edge $(i,j) \in \mathcal{E}$:

$$W_{ij} \mid A_{ij} = 1, \theta_{ij} \sim \mathcal{M}_V\left(M_{ij}, \beta^\top \theta_{ij}\right), \tag{6}$$

where $\beta_k = \text{softmax}(\rho^\top \alpha_k) \in \mathbb{R}^V$, $\beta = (\beta_1 \ldots \beta_K)^\top \in \mathcal{M}_{K\times V}(\mathbb{R})$, $\rho \in \mathcal{M}_{L\times V}(\mathbb{R})$, $\alpha_k \in \mathbb{R}^L$ and $\alpha = (\alpha_1 \ldots \alpha_K) \in \mathcal{M}_{L\times K}(\mathbb{R})$. Interestingly enough, discarding the network data and considering only the documents, the generative assumptions correspond to the embedded topic model (Dieng et al., 2020).

3 Inference

In the next section, the inference of the model is presented as well as the model selection criterion.

3.1 Variational inference and optimisation

In this work, we consider the marginal likelihood of the network and the texts for parameter estimation. The latent variables are denoted by $C = (C_i)_{i=1}^N$, $Z = (Z_i)_{i=1}^N$ and $Y = (Y_{ij})_{(i,j)\in\mathcal{E}}$. Moreover, the set of parameters $\Theta := \{\pi, \mu, \sigma, \kappa, m, s, \alpha, \rho\}$ is such that $m = (m_{qr})_{qr}$, $s = (s_{qr})_{qr}$, $\mu = (\mu_q)_q$ and $\sigma = (\sigma_q)_q$.Thus, the marginal log-likelihood is given by:

$$\mathcal{L}(\Theta; A, W) = \log p(A, W \mid \Theta) = \log\left(\sum_C \int_Z \int_Y p(A, W, C, Z, Y \mid \Theta) dZ dY\right). \tag{7}$$

Unfortunately, this quantity is not tractable since the sum over C requires to compute Q^N terms. Besides, it involves integrals that cannot be computed analytically. Therefore, we choose to rely on a variational inference approach for approximation purposes.

Decomposition of the marginal log-likelihood For any distribution $R(C, Z, Y)$, the following decomposition holds:

$$\mathcal{L}(\Theta; A, W) = \mathscr{L}(R(\cdot); \Theta) + \text{KL}(R(\cdot) || p(C, Z, Y \mid A, W)), \tag{8}$$

where

$$\mathscr{L}(R(\cdot);\Theta) = \mathbb{E}_R\left[\log \frac{p(A,W,C,Z,Y \mid \Theta)}{R(C,Z,Y)}\right]. \tag{9}$$

Since the Kullback-Leibler divergence is always positive in Equation (8), the ELBO $\mathscr{L}(R(\cdot);\Theta)$ is a lower bound of the marginal log-likelihood. Moreover, since the marginal log-likelihood does not depend on $R(\cdot)$, the higher the ELBO is, the closer to the marginal log-likelihood it is. To make the ELBO tractable, we restrict the family of variational distributions by considering a mean field assumption as well as the following hypothesis:

$$R(C,Z,Y) = R(C)R(Z)R(Y), \tag{10}$$

$$R_\tau(C) = \prod_{i=1}^{N} R_{\tau_i}(C_i) = \prod_{i=1}^{N} \mathcal{M}_Q(C_i;1,\tau_i), \tag{11}$$

$$R_{\phi_Z}(Z \mid A) = \prod_{i=1}^{N} R_{\phi_Z}(Z_i \mid A) = \prod_{i=1}^{N} \mathcal{N}(Z_i;\mu_{\phi_Z}(A)_i,\sigma^2_{\phi_Z}(A)_i I_p), \tag{12}$$

$$R_{\phi_Y}(Y \mid A,W) = \prod_{i\neq j} R_{\phi_Y}(Y_{ij} \mid W_{ij})^{A_{ij}} = \prod_{i\neq j} \mathcal{N}\big(Y_{ij};\mu_{\phi_Y}(W_{ij}),\operatorname{diag}\left(\sigma^2_{\phi_Y}(W_{ij})\right)\big)^{A_{ij}}, \tag{13}$$

where $\tau = (\tau_i)_{i=1}^N$ with $\forall i \in \{1,\dots,N\}$, $\tau_i \in \Delta_{Q-1}$. The notations $\mu_{\phi_Z}, \sigma^2_{\phi_Z}$ (μ_{ϕ_Y} and $\sigma^2_{\phi_Y}$respectively) denote the encoder of the nodes (the edges) and correspond to a mapping of the normalised adjacency matrix $\overline{A} := D^{-1/2}AD^{-1/2}$ (the document term matrix W) to the mean and standard deviation of the node (documents) representations into the latent space. The parameters of these encoders are denoted by ϕ_Z and ϕ_Y.

Optimisation To optimise the ELBO, we propose to alternate between closed form updates, based on first order conditions, and stochastic gradient descent steps using the Adam optimiser and the reparametrisation trick (Kingma, Welling, 2014; Rezende et al., 2014) that will be described in the talk, if the paper is accepted.

4 Numerical experiments

This section is dedicated to the assessment of the proposed methodology in this paper.

4.1 Simulation settings

To begin with, we introduce the three scenarios used to evaluate the model on different aspects, detailed hereinafter.

Scenarios

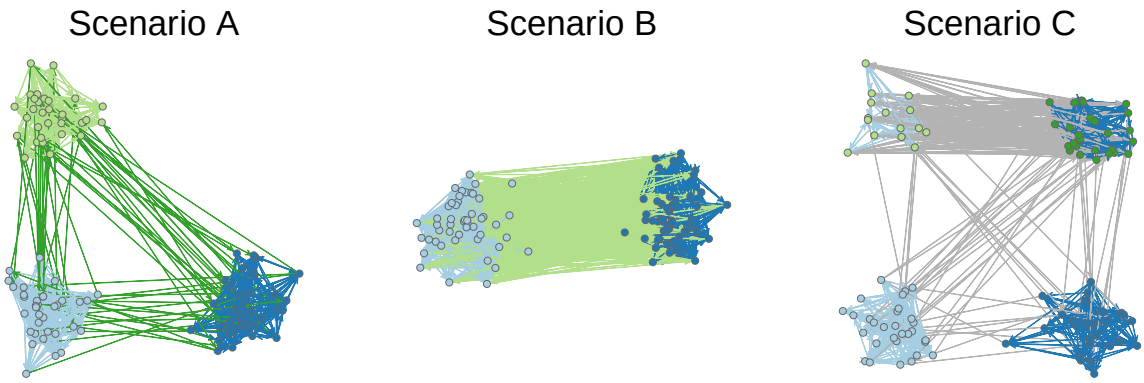

Figure 2: Representation of three networks simulated according to Scenario A, B and C.

- Scenario A is constituted of three communities, each defining a cluster, and four topics. By definition, a community is a group of nodes more densely connected together than with the rest of the network. For each cluster, a specific topic is employed to sample the documents associated with the intra-cluster connections. Besides, an extra topic is employed to model documents sent between nodes from different clusters. Hence, by construction, the clustering structure can be retrieved either using the network or the texts only.

- Scenario B is made of a single community and three topics. Consequently, all nodes connect with the same probability. Then, the nodes are spread into two clusters using distinct topics. An extra topic is used to model documents exchanged between the two clusters. Accordingly, the network itself is not sufficient to find the two clusters but the documents are.

- Scenario C comprises three communities and three topics. Two of the communities are associated with their respective topics, say t_1 and t_2. Furthermore, following Scenario B, the third community is split in two clusters, one being associated with topic t_1 and the other with t_2. Thus, considering both texts and topology, each network is actually made up of four node clusters. Consequently, both textual data and the network are necessary to uncover the clusters. This scenario will be of major interest in this experiment section since it ensures that the two sources of information are correctly used to uncover the partitions.

For all scenarios, the edges holding the documents are constructed by sampling words from four BBC articles, focusing each on a given topic. The first topic deals with the UK monarchy, the second with cancer treatments, the third with the political landscape in the UK and the last topic deals with astronomy. In the general setting, for all scenarios, the average text length for the documents is set to 150 words. The parameters used to sample data from the three scenarios illustrated in Figure 2, and given in Table 1. To summarise, the three proposed scenarios inspect different facets of the model. Scenario A insures that the model rightfully uses the network structure, Scenario B focuses on the usage of the topics to recover

the nodes partitions. Finally, Scenario C combines the two scenarios to guarantee that both sources of information are correctly utilised simultaneously.

	Scenario A	Scenario B	Scenario C
Q (clusters)	3	2	4
K (topics)	4	3	3
Communities	3	1	3
π_{qr} (connection probabilities) $\eta = 0.25$, $\epsilon = 0.01$	$\begin{pmatrix} \eta & \epsilon & \epsilon \\ \epsilon & \eta & \epsilon \\ \epsilon & \epsilon & \eta \end{pmatrix}$	$\begin{pmatrix} \eta & \eta \\ \eta & \eta \end{pmatrix}$	$\begin{pmatrix} \eta & \epsilon & \epsilon & \epsilon \\ \epsilon & \eta & \epsilon & \epsilon \\ \epsilon & \epsilon & \eta & \eta \\ \epsilon & \epsilon & \eta & \eta \end{pmatrix}$
Topics matrix $\mathbf{T}$ between pairs of clusters (q, r)	$\begin{pmatrix} t_1 & t_4 & t_4 \\ t_4 & t_2 & t_4 \\ t_4 & t_4 & t_3 \end{pmatrix}$	$\begin{pmatrix} t_1 & t_3 \\ t_3 & t_2 \end{pmatrix}$	$\begin{pmatrix} t_1 & t_3 & t_3 & t_3 \\ t_3 & t_2 & t_3 & t_3 \\ t_3 & t_3 & t_1 & t_3 \\ t_3 & t_3 & t_3 & t_2 \end{pmatrix}$

Table 1: Detail of the three simulation scenarios used to evaluate our model.

Clustering performance evaluation The adjusted rand index (ARI) is used as a measure of the closeness between two partitions. In this paper, ARI compares the true nodes labels with the nodes partition provided by the models. In particular, obtaining an ARI of 0 suggests that the clustering is as close to the true nodes labels as a random cluster assignment of the nodes. On the contrary, the closer the ARI is to 1, the better the results are. Ultimately, perfectly recovering the true partition (up to a label permutation) would lead to an ARI of 1.

Level of difficulty In order to generate more situations from the three scenarios, we introduce the *Hard* difficulty to test the model robustness against two aspects. First, we want to test the model against documents using several topics. Thus, in the *Hard* difficulty, the documents are formed of multiple topics such that, for any edge (i, j) with node i in cluster q and node j in cluster r, the topics proportions are computed as a ratio between the pure topic proportions $\theta^\star_{qr} \in \{0, 1\}^K$, with zeros everywhere except at the coordinate corresponding to the true topic and between the uniform distribution over the topics. This combination is controlled by a parameter ζ such that $\zeta = 0$ corresponds to a pure topic case while $\zeta = 1$ leads to a uniform distribution over the topics. This translates into:

$$\theta_{qr} = (1 - \zeta)\theta^\star_{qr} + \zeta * \left(\frac{1}{K}, \ldots, \frac{1}{K}\right)^\top, \tag{14}$$

with $\zeta = 0.7$ in the *Hard* setting. The second aspect tested by the *Hard* setting is the robustness in the presence of less connected communities. Consequently, the intra-cluster connection probability is decreased from $\eta = 0.25$ in the classical setting to $\eta = 0.1$ in the *Hard* one.

		ScenarioA	ScenarioB	ScenarioC
Easy	SBM	1.00 ± 0.00	-0.00 ± 0.01	0.73 ± 0.05
	STBM	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.01
	ETSBM	0.99 ± 0.03	1.00 ± 0.00	0.96 ± 0.04
	ETSBM - PT	1.00 ± 0.00	1.00 ± 0.00	0.96 ± 0.05
	Deep-LPTM	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
	Deep-LPTM - PT	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
Hard	SBM	0.97 ± 0.03	0.00 ± 0.00	0.62 ± 0.1
	STBM	0.63 ± 0.23	1.00 ± 0.00	0.66 ± 0.19
	ETSBM	0.96 ± 0.10	0.90 ± 0.30	0.72 ± 0.25
	ETSBM - PT	0.99 ± 0.01	1.00 ± 0.00	0.74 ± 0.21
	Deep-LPTM	0.99 ± 0.02	1.00 ± 0.00	0.89 ± 0.15
	Deep-LPTM - PT	1.00 ± 0.01	1.00 ± 0.00	0.85 ± 0.18

Table 2: ARI of the nodes clustering averaged over 10 graphs in all three scenarios for the two levels of difficulty Easy and Hard. Deep-LPTM, as well as ETSBM, are presented with and without pre-trained embeddings (denoted PT). Moreover, STBM and SBM are also provided as baselines.

4.2 Benchmark

In this section, we present a benchmark study in Table 2 comparing Deep-LPTM with state-of-the-art ETSBM and STBM. We also provide SBM as a baseline even though it cannot take into account the text edges. The table presents the average of the ARI over 10 graphs. Each graph result is obtained by running each method with five different initialisations and by taking the one resulting in the highest ELBO. The table is presented for four different models, namely SBM, STBM, ETSBM and Deep-LPTM. The last two models are evaluated with and without pre-trained embedding. In all cases, Deep-LPTM is either as good as or better than other models. In particular, in Scenario C with difficulty *Hard*, the ARI of Deep-LPTM node clustering is higher than all other methods, by at least 0.15. Likewise, in Scenario A with difficulty *Hard*, Deep-LPTM always recover the true partition while STBM only reaches an ARI of 0.66 ± 0.18.

Discussion In addition to the above, we derived a model selection criterion and analysed the Enron emails dataset. These aspects will also be discussed during the talk if the paper is accepted to the conference.

References

Dieng, Ruiz, Blei (2020). "Topic modeling in embedding spaces". In: *Transactions of the Association for Computational Linguistics* 8, pp. 439–453.

Handcock, Raftery, Tantrum (2007). "Model-based clustering for social networks". In: *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 170(2), pp. 301–354.

Kingma, Welling (2014). *Auto-Encoding Variational Bayes*.

Rezende, Mohamed, Wierstra (2014). "Stochastic backpropagation and approximate inference in deep generative models". In: *International conference on machine learning*. PMLR, pp. 1278–1286.