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

Rémi Boutin[1], Pierre Latouche[2,1] and Charles Bouveyron[3]

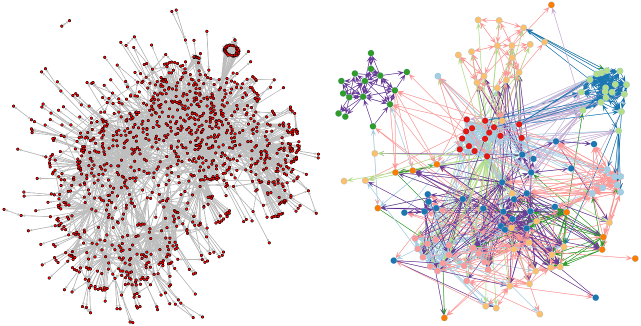
[1] MAP5 - Université de Paris Cité
[2] LMBP - Université Clermont Auvergne
[3] Maasai team - INRIA, Université Côte d'Azur

JDS 2023

Introduction

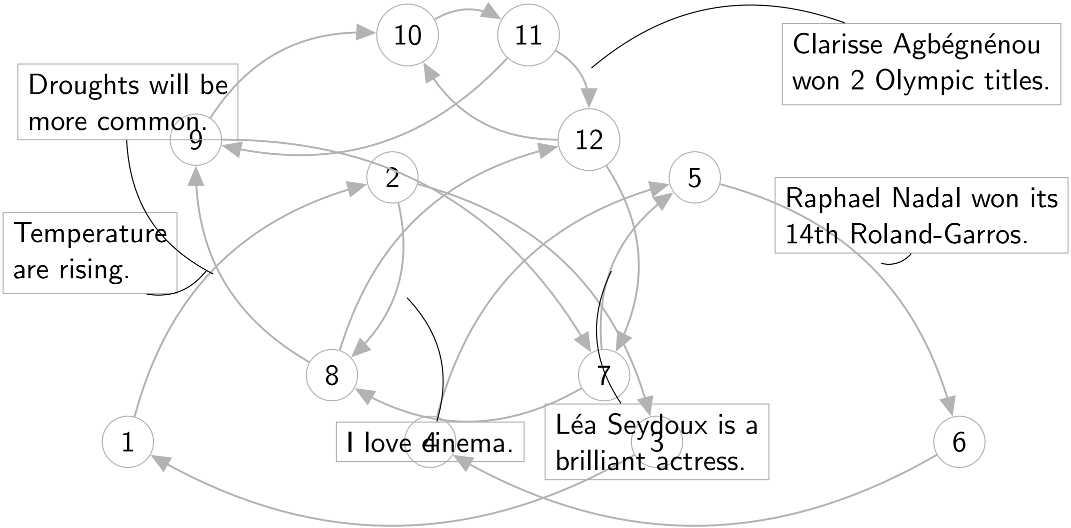
Networks can be observed directly or indirectly from a variety of sources:

- social websites (Facebook, Twitter, ...),
- emails (from your Gmail, Clinton's mails, Enron Email data ...),
- digital/numeric documents (Panama papers, co-authorships, ...),
- and even archived documents in libraries (digital humanities).

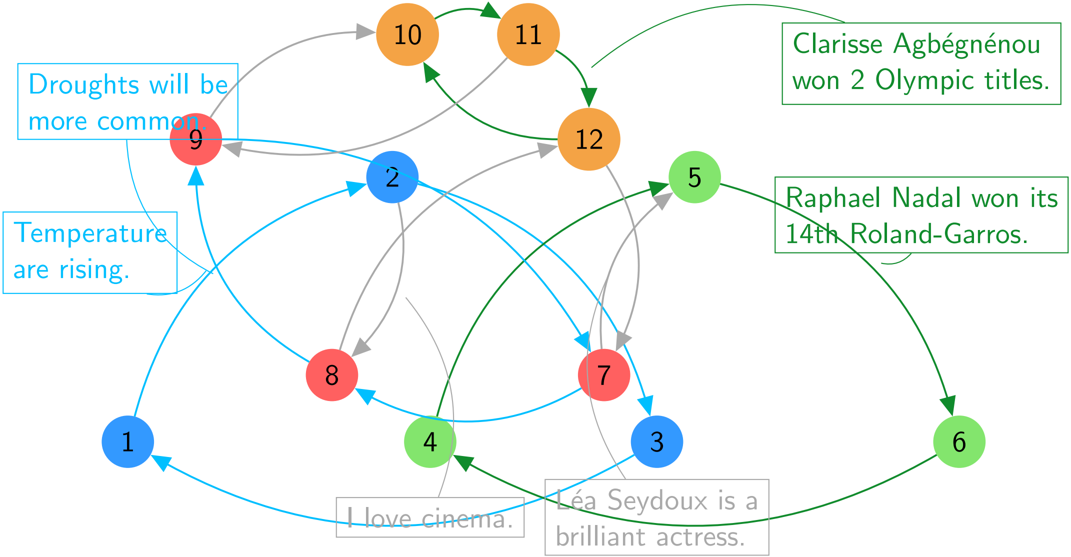


⇒ **most of these sources involve text!**

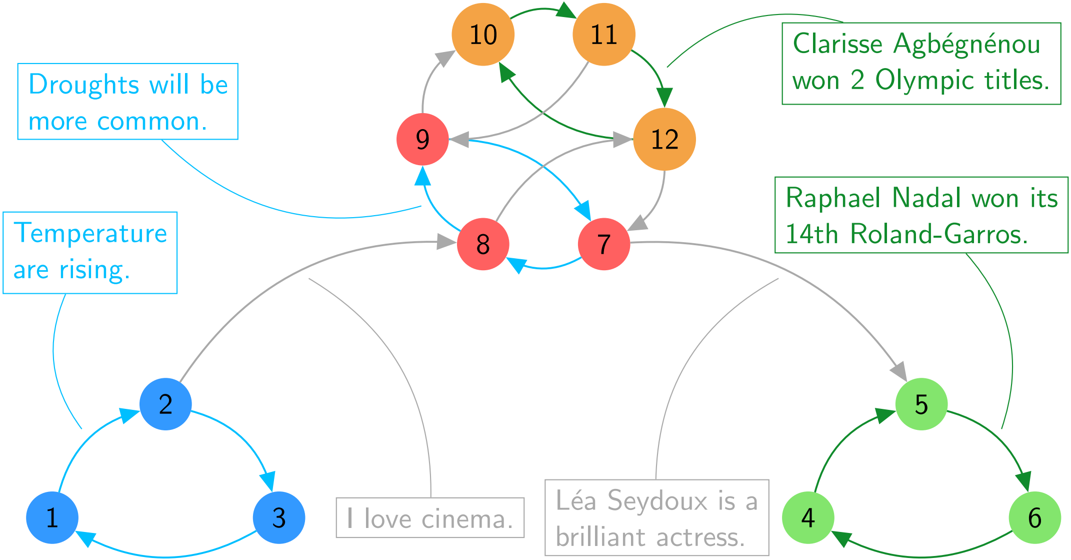
Observed network: difficult to apprehend



STBM/ETSBM results: difficult to represent



Our goal with Deep-LPTM



Node generation

Based on the latent position cluster model (Handcock et al., 2007), C_i the cluster membership of node i for all $i \in \{1, \ldots, M\}$

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

where Q corresponds to the number of clusters. The latent vector representing node i, denoted Z_i, is assumed to be Gaussian:

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

Denoting $\eta_{ij} := \kappa - \|Z_i - Z_j\|$, the probability for node i to be connected to node j is

$$P(A_{ij} = 1 \mid Z_i, Z_j, \kappa) = \frac{1}{1 + e^{-\eta_{ij}}}. \tag{3}$$

Node generation

Based on the latent position cluster model (Handcock et al., 2007), C_i the cluster membership of node i for all $i \in \{1, \ldots, M\}$

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

where Q corresponds to the number of clusters. The latent vector representing node i, denoted Z_i, is assumed to be Gaussian:

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

Denoting $\eta_{ij} := \kappa - \|Z_i - Z_j\|$, the probability for node i to be connected to node j is

$$P(A_{ij} = 1 \mid Z_i, Z_j, \kappa) = \frac{1}{1 + e^{-\eta_{ij}}}. \tag{3}$$

Node generation

Based on the latent position cluster model (Handcock et al., 2007), C_i the cluster membership of node i for all $i \in \{1, \ldots, M\}$

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

where Q corresponds to the number of clusters. The latent vector representing node i, denoted Z_i, is assumed to be Gaussian:

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

Denoting $\eta_{ij} := \kappa - \|Z_i - Z_j\|$, the probability for node i to be connected to node j is

$$P(A_{ij} = 1 \mid Z_i, Z_j, \kappa) = \frac{1}{1 + e^{-\eta_{ij}}}. \tag{3}$$

Node generation

Based on the latent position cluster model (Handcock et al., 2007), C_i the cluster membership of node i for all $i \in \{1, \ldots, M\}$

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

where Q corresponds to the number of clusters. The latent vector representing node i, denoted Z_i, is assumed to be Gaussian:

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

Denoting $\eta_{ij} := \kappa - \|Z_i - Z_j\|$, the probability for node i to be connected to node j is

$$P(A_{ij} = 1 \mid Z_i, Z_j, \kappa) = \frac{1}{1 + e^{-\eta_{ij}}}. \tag{3}$$

Text generation in Deep-LPTM

1. $Y_{ij} \mid A_{ij} C_{iq} C_{jr} = 1 \sim \mathcal{N}_K(m_{qr}, s_{qr}^2 I_K)$,
2. $\theta_{ij} = \mathrm{softmax}(Y_{ij})$, proportions of topic in documents sent from i to j
3. $W_{ij} \mid A_{ij} = 1, \theta_{ij} \sim \mathcal{M}_V\left(M_{ij}, \beta^\top \theta_{ij}\right)$, where $\beta = (\beta_1, \ldots, \beta_K)^\top \in \mathcal{M}_{K \times V}((0,1))$ and

$$\beta_k = \mathrm{softmax}\left(\rho^\top \alpha_k\right).$$

vocabulary $\in \mathbb{R}^{V \times L}$ topic vector $\in \mathbb{R}^L$

This is based on ETM (Dieng et al., 2020), allowing to use pre-trained embeddings to incorporate semantic meaning.

Text generation in Deep-LPTM

1. $Y_{ij} \mid A_{ij}C_{iq}C_{jr} = 1 \sim \mathcal{N}_K(m_{qr}, s_{qr}^2 I_K)$,
2. $\theta_{ij} = \mathrm{softmax}(Y_{ij})$, proportions of topic in documents sent from i to j
3. $W_{ij} \mid A_{ij} = 1, \theta_{ij} \sim \mathcal{M}_V\left(M_{ij}, \beta^\top \theta_{ij}\right)$, where $\beta = (\beta_1, \ldots, \beta_K)^\top \in \mathcal{M}_{K\times V}((0,1))$ and

$$\beta_k = \mathrm{softmax}\left(\rho^\top \alpha_k\right).$$

vocabulary $\in \mathbb{R}^{V\times L}$ topic vector $\in \mathbb{R}^L$

This is based on ETM (Dieng et al., 2020), allowing to use pre-trained embeddings to incorporate semantic meaning.

Text generation in Deep-LPTM

1. $Y_{ij} \mid A_{ij}C_{iq}C_{jr} = 1 \sim \mathcal{N}_K(m_{qr}, s_{qr}^2 I_K)$,
2. $\theta_{ij} = \mathrm{softmax}(Y_{ij})$, proportions of topic in documents sent from i to j
3. $W_{ij} \mid A_{ij} = 1, \theta_{ij} \sim \mathcal{M}_V \left(M_{ij}, \beta^\top \theta_{ij}\right)$, where $\beta = (\beta_1, \ldots, \beta_K)^\top \in \mathcal{M}_{K\times V}\left((0,1)\right)$ and

$$\beta_k = \mathrm{softmax}\left(\underbrace{\rho^\top}_{\text{vocabulary} \in \mathbb{R}^{V\times L}} \underbrace{\alpha_k}_{\text{topic vector} \in \mathbb{R}^{L}}\right).$$

This is based on ETM (Dieng et al., 2020), allowing to use pre-trained embeddings to incorporate semantic meaning.

Inference

- C_i node cluster membership
- Z_i node latent representation
- Y_{ij} text latent representation

Marginal likelihood

Denoting Θ the set of all model parameters,

$$\log p(A, W \mid \Theta) = \log \left(\sum_{C} \int_{Z} \int_{Y} p(A, W, C, Z, Y \mid \Theta) dZ dY \right). \tag{4}$$

This quantity is not tractable since the sum over all configurations requires to compute Q^N terms. Besides, it involves integrals that cannot be computed analytically.

$\longrightarrow$ **Variational inference** for approximation purposes.

Inference

- C_i node cluster membership
- Z_i node latent representation
- Y_{ij} text latent representation

Marginal likelihood

Denoting Θ the set of all model parameters,

$$\log p(A, W \mid \Theta) = \log \left(\sum_{C} \int_{Z} \int_{Y} p(A, W, C, Z, Y \mid \Theta) dZ dY \right). \tag{4}$$

This quantity is not tractable since the sum over all configurations requires to compute Q^N terms. Besides, it involves integrals that cannot be computed analytically.

$\longrightarrow$ **Variational inference** for approximation purposes.

Inference

- C_i node cluster membership
- Z_i node latent representation
- Y_{ij} text latent representation

Marginal likelihood

Denoting Θ the set of all model parameters,

$$\log p(A, W \mid \Theta) = \log \left(\sum_{C} \int_{Z} \int_{Y} p(A, W, C, Z, Y \mid \Theta) dZ dY \right). \tag{4}$$

This quantity is not tractable since the sum over all configurations requires to compute Q^N terms. Besides, it involves integrals that cannot be computed analytically.

$\longrightarrow$ **Variational inference** for approximation purposes.

Inference

- C_i node cluster membership
- Z_i node latent representation
- Y_{ij} text latent representation

The **v ariational inference** consists in splitting the likelihood in two terms. For any distribution $R(C, Z, Y)$,

$$\log p(A, W \mid \Theta) = \mathscr{L}(R(\cdot); \Theta) + \mathrm{KL}(R(\cdot) || p(C, Z, Y \mid A, W)), \tag{5}$$

where

$$\mathscr{L}(R(\cdot); \Theta) = \mathbb{E}_R \left[\log \frac{p(A, W, C, Z, Y \mid \Theta)}{R(C, Z, Y)} \right]. \tag{6}$$

Inference

- C_i node cluster membership
- Z_i node latent representation
- Y_{ij} text latent representation

The **v ariational inference** consists in splitting the likelihood in two terms. For any distribution $R(C, Z, Y)$,

$$\log p(A, W \mid \Theta) = \mathscr{L}(R(\cdot); \Theta) + \mathrm{KL}(R(\cdot) || p(C, Z, Y \mid A, W)), \tag{5}$$

where

$$\mathscr{L}(R(\cdot); \Theta) = \mathbb{E}_R \left[\log \frac{p(A, W, C, Z, Y \mid \Theta)}{R(C, Z, Y)} \right]. \tag{6}$$

Inference

- C_i node cluster membership
- Z_i node latent representation
- Y_{ij} text latent representation

Assumptions regarding the variational distributions:

$$R(C, Z, Y \mid A, W) = R(C)R(Z \mid A)R(Y \mid A, W),$$

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

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

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

where $\left(\mu_{\phi_Z}, \sigma^2_{\phi_Z}\right)$ are the outputs of the encoder of a variational graph auto encoder (Kipf, Welling, 2016) and $\left(\mu_{\phi_Y}, \sigma^2_{\phi_Y}\right)$ the outputs of ETM encoder.

Inference

- C_i node cluster membership
- Z_i node latent representation
- Y_{ij} text latent representation

Assumptions regarding the variational distributions:

$$R(C, Z, Y \mid A, W) = R(C)R(Z \mid A)R(Y \mid A, W),$$

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

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

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

where $\left(\mu_{\phi_Z}, \sigma^2_{\phi_Z}\right)$ are the outputs of the encoder of a variational graph auto encoder (Kipf, Welling, 2016) and $\left(\mu_{\phi_Y}, \sigma^2_{\phi_Y}\right)$ the outputs of ETM encoder.

Inference

- C_i node cluster membership
- Z_i node latent representation
- Y_{ij} text latent representation

Assumptions regarding the variational distributions:

$$R(C, Z, Y \mid A, W) = R(C)R(Z \mid A)R(Y \mid A, W),$$

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

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

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

where $\left(\mu_{\phi_Z}, \sigma^2_{\phi_Z}\right)$ are the outputs of the encoder of a variational graph auto encoder (Kipf, Welling, 2016) and $\left(\mu_{\phi_Y}, \sigma^2_{\phi_Y}\right)$ the outputs of ETM encoder.

Inference

- C_i node cluster membership
- Z_i node latent representation
- Y_{ij} text latent representation

Assumptions regarding the variational distributions:

$$R(C, Z, Y \mid A, W) = R(C)R(Z \mid A)R(Y \mid A, W),$$

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

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

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

where $\left(\mu_{\phi_Z}, \sigma^2_{\phi_Z}\right)$ are the outputs of the encoder of a variational graph auto encoder (Kipf, Welling, 2016) and $\left(\mu_{\phi_Y}, \sigma^2_{\phi_Y}\right)$ the outputs of ETM encoder.

Inference

- C_i node cluster membership
- Z_i node latent representation
- Y_{ij} text latent representation

Assumptions regarding the variational distributions:

$$R(C, Z, Y \mid A, W) = R(C)R(Z \mid A)R(Y \mid A, W),$$

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

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

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

where $\left(\mu_{\phi_Z}, \sigma^2_{\phi_Z}\right)$ are the outputs of the encoder of a variational graph auto encoder (Kipf, Welling, 2016) and $\left(\mu_{\phi_Y}, \sigma^2_{\phi_Y}\right)$ the outputs of ETM encoder.

Inference

- C_i node cluster membership
- Z_i node latent representation
- Y_{ij} text latent representation

Optimisation of the ELBO

- analytical updates for μ, σ, m and s
- stochastic gradient descent for κ, ϕ_Z and ϕ_Y

Synthetic datasets

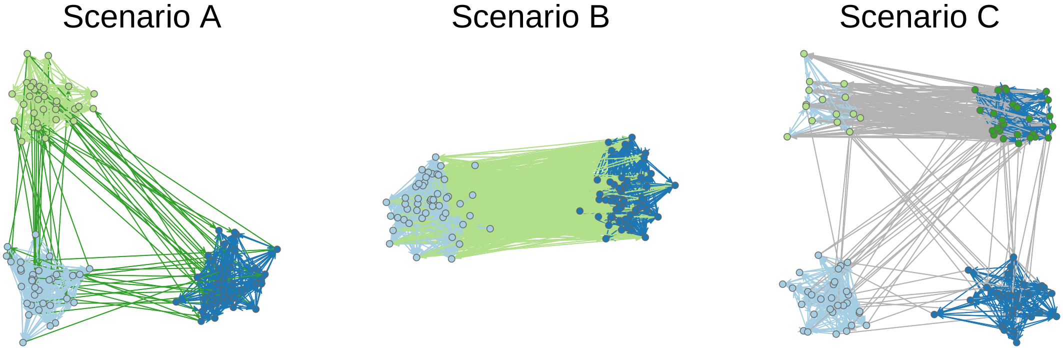
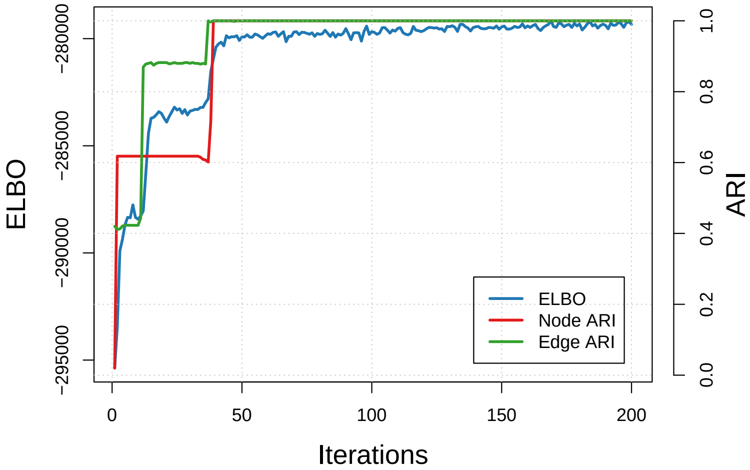


Figure: Networks sampled from each scenario. The node colours denote the node cluster memberships and the edge colours denote the majority topic in the corresponding documents.

Simulations - Detailed example with three communities



Figure

Benchmark

		ScenarioA	ScenarioB	ScenarioC
Easy	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	ETSBM	0.96 ± 0.10	0.90 ± 0.30	0.72 ± 0.25
	ETSBM - PT	0.99 ± 0.01	$\mathbf{1.00} \pm \mathbf{0.00}$	0.74 ± 0.21
	Deep-LPTM	0.99 ± 0.02	$\mathbf{1.00} \pm \mathbf{0.00}$	$\mathbf{0.89} \pm \mathbf{0.15}$
	Deep-LPTM - PT	$\mathbf{1.00} \pm \mathbf{0.01}$	$\mathbf{1.00} \pm \mathbf{0.00}$	0.85 ± 0.18

Table: ARI of the node clustering over 10 graphs in 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)

Real world example: ENRON email dataset

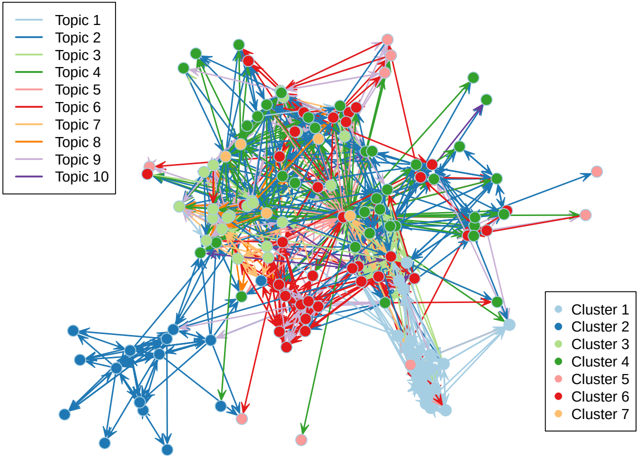


Figure: Deep-LPTM representation of Enron email network. The node cluster memberships are denoted by the colour of the nodes and the majority topic in the documents are denoted by the colour of the edges.

Real world example: ENRON email dataset

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
tw	ercot	rto	backup	ofo
watson	vepco	steffes	seat	interview
hayslett	ene	christi	location	cycle
donoho	liz	nicolay	test	mmbtu
lindy	dyn	novosel	supplies	usage
geaccone	filename	affairs	building	interviewers
lynn	mws	rtos	floors	fantastic
transwestern	desk	shapiro	mails	super
teb	mw	government	notified	deliveries
lohman	enpower	skilling	seats	dinner

Topic 6	Topic 7	Topic 8	Topic 9	Topic 10
sara	frontier	grigsby	master	edison
shackleton	western	desk	nymex	puc
kim	williams	mike	handling	dwr
ward	dt	taleban	isda	davis
master	project	forces	executed	dasovich
isda	whitt	sheppard	agreement	sce
perlingiere	dth	afghanistan	netting	da
perlingiereenron	enw	holst	multicurrency	state
leathercenter	marathon	gaskill	na	california
shackletonenron	cheyenne	ina	cn	jeff

Figure: The 10 most probable words of each topic according to Deep-LPTM.

Conclusion & further work

- The representation for communities works fine
- The clustering is efficient in the three studied settings
- Our model captures meaningful clusters both in terms of connections and topics
- Combining the block modelling approach with the representation power
- Improve the graph neural network with latest advancement
- Incorporate temporal information

Bibliography

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.

Kipf, Welling (2016). "Variational graph auto-encoders". In: *arXiv preprint arXiv:1611.07308*.

Email: remi.boutin.stat@gmail.com