Deep graphical models and inference strategies for the analysis of networks comprising textual edges

PhD defence

Rémi Boutin

Directors: Pr. Pierre Latouche and Pr. Charles Bouveyron

Thursday 14th december 2023

Laboratoire MAP5, Université Paris Cité

Introduction

The rise of complex networks

- Surge of the storage capacity
- Diversification of data sources (e.g. texts, images, quantitative variables)

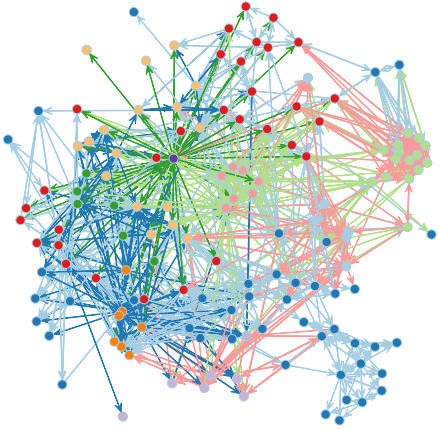


Figure 1: An email network from Bouveyron, Latouche, and Zreik (2018).

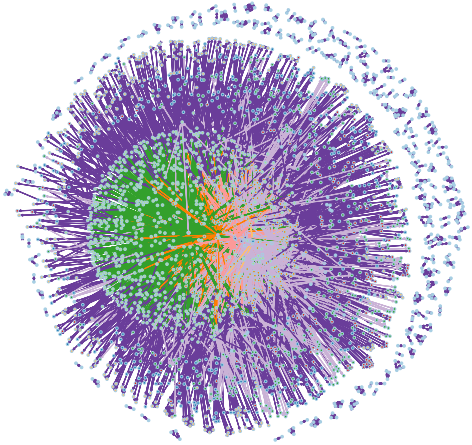
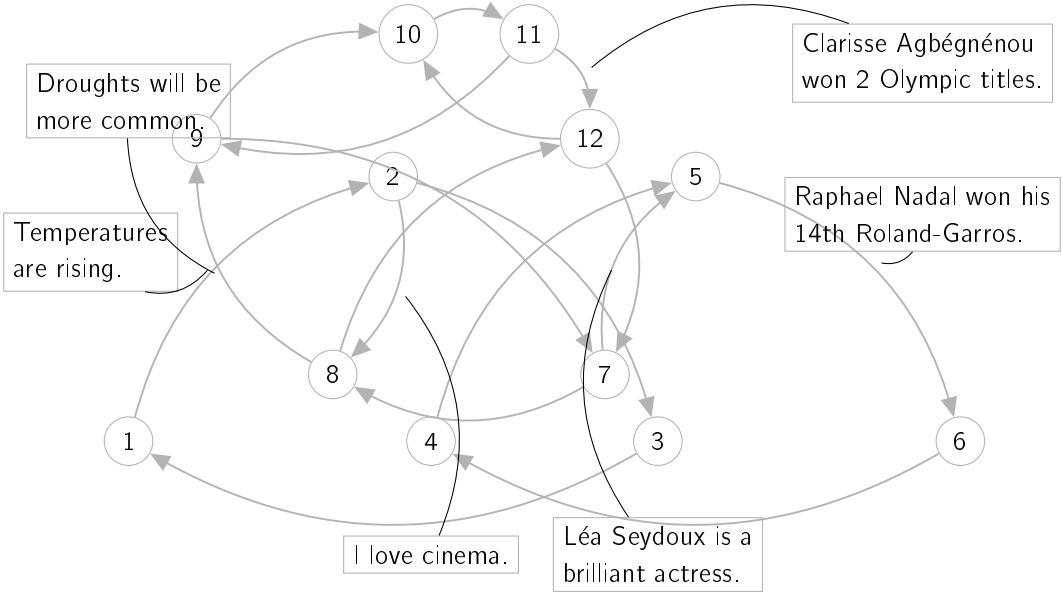
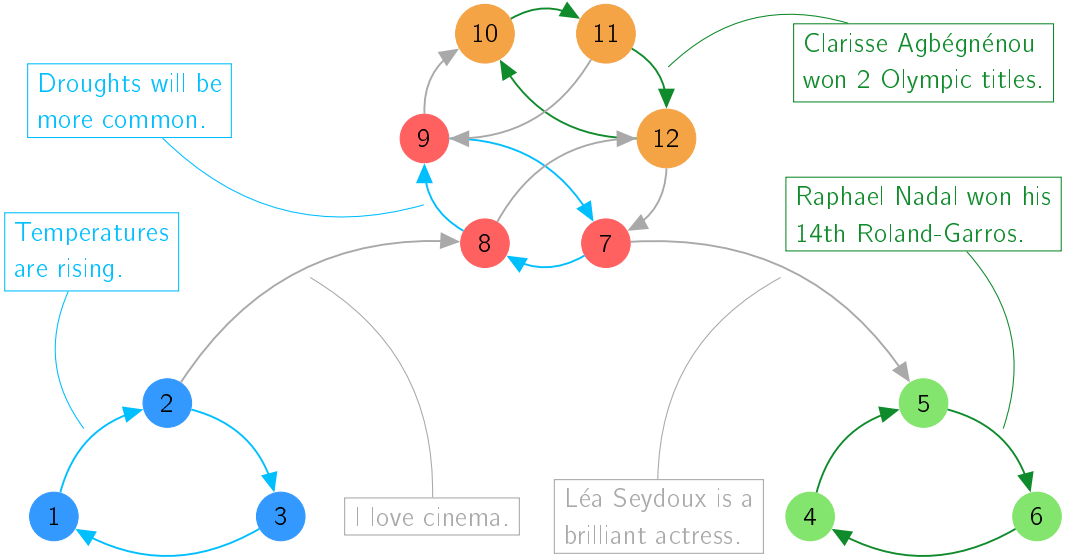


Figure 2: A co-authorship network from Bouveyron, Latouche, and Zreik (2018).

Example of the necessity to capture underlying patterns of a network

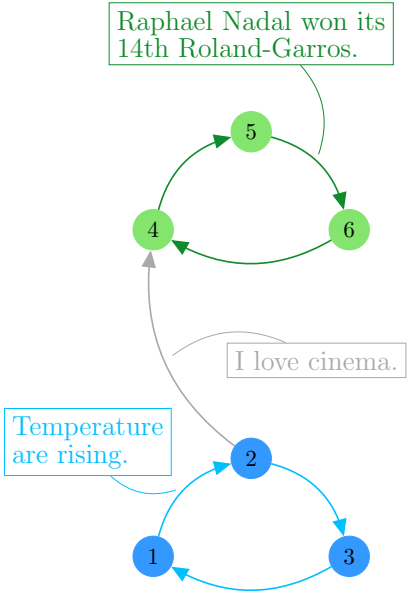


Example of the necessity to capture underlying patterns of a network



Notations

- i and j will refer to **nodes**.
- q and r will refer to **clusters**.
- $\beta_k \in \Delta_V$: **a topic** over the V words.
- k will refer to **topics**.
- Q: the **number of clusters**.
- K: the **number of topics**.
- N: the **number of nodes**.
- M: the **number of edges**.
- $\text{softmax}(x) = (\sum_{k=1}^{K} e^{x_k})^{-1}(e^{x_1}, \ldots, e^{x_K})$, $\forall x \in \mathbb{R}^K$.



Objective of this thesis

Main objective of this thesis: develop new methodologies to improve node clustering and topic modelling on a network with textual edges to improve the relevancy of the results.

- Can advancements in neural topic models help to improve clustering ?

→ First, we propose the embedded topics in the stochastic block model.

- Can deep encoding of the documents and of the connections help to solve node clustering as well as topic modelling ?
- How to estimate node positions for visualisation purposes in an end-to-end methodology ?

→ Second, we present the deep latent position topic model.

Finally, we discard the textual edges to focus on the connectivity.

- Can block modelling and position estimation be performed simultaneously, based on a deep probabilistic model ?

→ The deep latent positional block model is introduced in the third section.

Objective of this thesis

Main objective of this thesis: develop new methodologies to improve node clustering and topic modelling on a network with textual edges to improve the relevancy of the results.

- Can advancements in neural topic models help to improve clustering ?

→ First, we propose the embedded topics in the stochastic block model.

- Can deep encoding of the documents and of the connections help to solve node clustering as well as topic modelling ?
- How to estimate node positions for visualisation purposes in an end-to-end methodology ?

→ Second, we present the deep latent position topic model.

Finally, we discard the textual edges to focus on the connectivity.

- Can block modelling and position estimation be performed simultaneously, based on a deep probabilistic model ?

→ The deep latent positional block model is introduced in the third section.

Objective of this thesis

Main objective of this thesis: develop new methodologies to improve node clustering and topic modelling on a network with textual edges to improve the relevancy of the results.

- Can advancements in neural topic models help to improve clustering ?

→ First, we propose the embedded topics in the stochastic block model.

- Can deep encoding of the documents and of the connections help to solve node clustering as well as topic modelling ?
- How to estimate node positions for visualisation purposes in an end-to-end methodology ?

→ Second, we present the deep latent position topic model.

Finally, we discard the textual edges to focus on the connectivity.

- Can block modelling and position estimation be performed simultaneously, based on a deep probabilistic model ?

→ The deep latent positional block model is introduced in the third section.

Embedded topics in the stochastic block model

1. Embedded topics in the stochastic block model
2. Deep latent position topic model
3. Deep latent position block model
4. Conclusion and perspectives

Generative model: Assumptions about the generation of networks with textual edges

- Node cluster memberships:

$$C_i \mid \gamma \overset{\text{i.i.d}}{\sim} \mathcal{M}_Q(1, \gamma).$$

- Connection between nodes i and j:

$$A_{ij} \mid \{C_{iq}C_{jr} = 1, \pi_{qr}\} \overset{\text{i.i.d}}{\sim} \mathcal{B}(\pi_{qr}).$$

- Topic proportions:

$$Y_{qr} \overset{\text{i.i.d}}{\sim} \mathcal{N}_K(0_K, \mathbf{I}_K) \text{ and } \theta_{qr} = \text{softmax}(Y_{qr}).$$

- Word within documents sent from i to j[1]:

$$W_{ij}^n \mid \{C_{iq}C_{jr}A_{ij} = 1, \theta_{qr}, \boldsymbol{\alpha}, \boldsymbol{\rho}\} \sim \mathcal{M}_V(1, \theta_{qr}^\top \boldsymbol{\beta}).$$

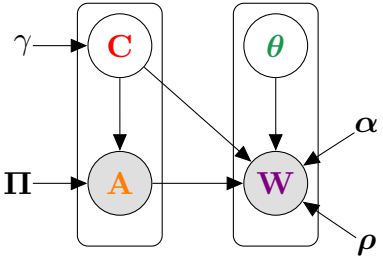


Figure 3: Graphical representation of ETSBM.

[1] Dieng, Ruiz, and Blei (2020)

Generative model: a decoder based on word and topic embeddings

$$W_{ij}^n \mid \{C_{iq}C_{jr}A_{ij} = 1, \theta_{qr}, \boldsymbol{\alpha}, \boldsymbol{\rho}\} \sim \mathcal{M}_V(1, {\theta_{qr}}^\top \boldsymbol{\beta}),$$

where $\boldsymbol{\beta} = (\beta_1 \cdots \beta_K)^\top \in \mathcal{M}_{K \times V}(\mathbb{R})$ is the vocabulary matrix and

$${\theta_{qr}}^\top \boldsymbol{\beta} = \sum_{k=1}^{K} \theta_{qrk} \boldsymbol{\beta}_{k,\cdot} \in \mathbb{R}^V.$$

Each topic k, represented by $\beta_k \in \Delta_V$, is obtained by computing:

$$\beta_k = \text{softmax}\left(\boldsymbol{\rho}^\top \alpha_k\right),$$

- $\boldsymbol{\rho} \in \mathcal{M}_{L \times V}(\mathbb{R})$ L-dimensional word embeddings
- $\boldsymbol{\alpha} = (\alpha_1 \cdots \alpha_K) \in \mathcal{M}_{L \times K}(\mathbb{R})$ L-dimensional topic embeddings

Inference: Bayesian framework

Using a Bayesian framework provides a reliable model selection criterion[2].

Assumed a priori distributions:

$$\gamma \sim \mathcal{D}ir_Q(\gamma_0),$$
$$\pi_{qr} \overset{\text{i.i.d}}{\sim} \mathcal{B}eta(a, b).$$

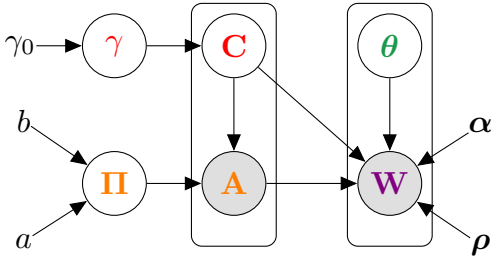


Figure 4: Graphical representation of ETSBM.

[2] Latouche, Birmele, and Ambroise (2012); McDaid et al. (2013); Côme and Latouche (2015).

Inference: the integrated log-likelihood

The integrated joint log-likelihood is given by:

Inference: the integrated log-likelihood

The integrated joint log-likelihood is given by:

$$\log p(\mathbf{A}, \mathbf{W} \mid \boldsymbol{\alpha}, \boldsymbol{\rho}) = \log \left(\sum_{\mathbf{C}} \int_{\mathbf{Y}} \int_{\gamma} \int_{\mathbf{\Pi}} p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{\Pi}, \gamma, \mathbf{Y} \mid \boldsymbol{\alpha}, \boldsymbol{\rho}) d\mathbf{\Pi} d\mathbf{Y} d\gamma \right).$$

Inference: the integrated log-likelihood

The integrated joint log-likelihood is given by:

$$\log p(\mathbf{A}, \mathbf{W} \mid \boldsymbol{\alpha}, \boldsymbol{\rho}) = \log \left(\sum_{\mathbf{C}} \int_{\mathbf{Y}} \int_{\gamma} \int_{\mathbf{\Pi}} p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{\Pi}, \gamma, \mathbf{Y} \mid \boldsymbol{\alpha}, \boldsymbol{\rho}) d\mathbf{\Pi} d\mathbf{Y} d\gamma \right).$$

Unfortunately, this is not tractable:

Inference: the integrated log-likelihood

The integrated joint log-likelihood is given by:

$$\log p(\mathbf{A}, \mathbf{W} \mid \boldsymbol{\alpha}, \boldsymbol{\rho}) = \log \left(\sum_{\mathbf{C}} \int_{\mathbf{Y}} \int_{\gamma} \int_{\mathbf{\Pi}} p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{\Pi}, \gamma, \mathbf{Y} \mid \boldsymbol{\alpha}, \boldsymbol{\rho}) d\mathbf{\Pi} d\mathbf{Y} d\gamma \right).$$

Unfortunately, this is not tractable:

- sum over Q^N configurations of $\mathbf{C}$, **exponential in the number of nodes !**

Inference: the integrated log-likelihood

The integrated joint log-likelihood is given by:

$$\log p(\mathbf{A}, \mathbf{W} \mid \boldsymbol{\alpha}, \boldsymbol{\rho}) = \log \left(\sum_{\mathbf{C}} \int_{\mathbf{Y}} \int_{\gamma} \int_{\mathbf{\Pi}} p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{\Pi}, \gamma, \mathbf{Y} \mid \boldsymbol{\alpha}, \boldsymbol{\rho}) d\mathbf{\Pi} d\mathbf{Y} d\gamma \right).$$

Unfortunately, this is not tractable:

- sum over Q^N configurations of $\mathbf{C}$, **exponential in the number of nodes !**
- the $\mathrm{softmax}$ function prevents from obtaining any analytical expression

Inference: variational approximation

1. **EM algorithm ?** $p(C_i \mid \mathbf{A})$ cannot be reduced to $P(C_i \mid \mathbf{A}_i)$[3].
2. **Variational inference:**

Denoting $R(\cdot)$, a distribution on $\mathbf{C}, \mathbf{\Pi}, \gamma$ and $\mathbf{Y}$, the integrated joint log-likelihood can be decomposed as follow:

$$\log p(\mathbf{A}, \mathbf{W} \mid \boldsymbol{\alpha}, \boldsymbol{\rho}) = \mathscr{L}(R(\cdot); \boldsymbol{\alpha}, \boldsymbol{\rho}) + \mathrm{KL}(R(\cdot) || p(\mathbf{C}, \mathbf{\Pi}, \gamma, \mathbf{Y} \mid \mathbf{A}, \mathbf{W}, \boldsymbol{\alpha}, \boldsymbol{\rho})),$$

where

$$\mathscr{L}(R(\cdot); \boldsymbol{\alpha}, \boldsymbol{\rho}) = \mathbb{E}_R \left[\log \frac{\overbrace{p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{\Pi}, \gamma, \mathbf{Y} \mid \boldsymbol{\alpha}, \boldsymbol{\rho})}^{\text{full joint distribution}}}{\underbrace{R(\mathbf{C}, \mathbf{\Pi}, \gamma, \mathbf{Y})}_{\text{variational distribution}}} \right].$$

[3] Daudin, Picard, and Robin (2008).

1. **EM algorithm ?** $p(C_i \mid \mathbf{A})$ cannot be reduced to $P(C_i \mid \mathbf{A}_i)$[3].
2. **Variational inference:**

Denoting $R(\cdot)$, a distribution on $\mathbf{C}, \mathbf{\Pi}, \gamma$ and $\mathbf{Y}$, the integrated joint log-likelihood can be decomposed as follow:

$$\log p(\mathbf{A}, \mathbf{W} \mid \boldsymbol{\alpha}, \boldsymbol{\rho}) = \mathscr{L}(R(\cdot); \boldsymbol{\alpha}, \boldsymbol{\rho}) + \mathrm{KL}(R(\cdot) || p(\mathbf{C}, \mathbf{\Pi}, \gamma, \mathbf{Y} \mid \mathbf{A}, \mathbf{W}, \boldsymbol{\alpha}, \boldsymbol{\rho})),$$

where

$$\mathscr{L}(R(\cdot); \boldsymbol{\alpha}, \boldsymbol{\rho}) = \mathbb{E}_R \left[\log \frac{\overbrace{p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{\Pi}, \gamma, \mathbf{Y} \mid \boldsymbol{\alpha}, \boldsymbol{\rho})}^{\text{full joint distribution}}}{\underbrace{R(\mathbf{C}, \mathbf{\Pi}, \gamma, \mathbf{Y})}_{\text{variational distribution}}} \right].$$

[3] Daudin, Picard, and Robin (2008).

Inference: variational distribution assumptions

To make $\mathscr{L}(R(\cdot); \boldsymbol{\alpha}, \boldsymbol{\rho})$ tractable, we use the following mean-field assumption:

$$R(\mathbf{C}, \boldsymbol{\Pi}, \gamma, \mathbf{Y}) = R(\mathbf{C})R(\boldsymbol{\Pi})R(\gamma)R\big(\mathbf{Y}\big).$$

Inference: variational distribution assumptions

To make $\mathscr{L}(R(\cdot); \boldsymbol{\alpha}, \boldsymbol{\rho})$ tractable, we use the following mean-field assumption:

$$R(\mathbf{C}, \boldsymbol{\Pi}, \gamma, \mathbf{Y}) = R(\mathbf{C})R(\boldsymbol{\Pi})R(\gamma)R\big(\mathbf{Y}\big).$$

We also assume the following variational distributions:

$$\begin{aligned}
R(\mathbf{C}) &= \prod_{i=1}^{M} R(C_i) = \prod_{i=1}^{M} \mathcal{M}_Q(C_i; 1, \tau_i), \\
R(\boldsymbol{\Pi}) &= \prod_{q,r=1}^{Q} R(\pi_{qr}) = \prod_{q,r=1}^{Q} \mathcal{B}eta(\pi_{qr}; \tilde{\pi}_{qr1}, \tilde{\pi}_{qr2}), \\
R(\gamma) &= \mathcal{D}ir_Q(\gamma; \tilde{\gamma}).
\end{aligned}$$

Inference: meta-documents construction

Regarding the latent topic proportions θ_{qr}, we propose a new encoding of textual data:

$$R(\mathbf{Y}) = \prod_{q,r=1}^{Q} R(Y_{qr}) = \prod_{q,r=1}^{Q} \mathcal{N}_K\left(Y_{qr}; \mu_{qr}^{\nu}(\boldsymbol{\tau}), \mathrm{diag}((\sigma_{qr}^{\nu})^2(\boldsymbol{\tau}))\right),$$

Inference: meta-documents construction

Regarding the latent topic proportions θ_{qr}, we propose a new encoding of textual data:

$$R(\mathbf{Y}) = \prod_{q,r=1}^{Q} R(Y_{qr}) = \prod_{q,r=1}^{Q} \mathcal{N}_K\left(Y_{qr}; \mu_{qr}^{\nu}(\boldsymbol{\tau}), \mathrm{diag}((\sigma_{qr}^{\nu})^2(\boldsymbol{\tau}))\right),$$

- $W_{qr} = \{W_{ij} : C_{iq}C_{jr} = 1\}$, the set of documents sent from cluster q to cluster r.

.

Inference: meta-documents construction

Regarding the latent topic proportions θ_{qr}, we propose a new encoding of textual data:

$$R(\mathbf{Y}) = \prod_{q,r=1}^{Q} R(Y_{qr}) = \prod_{q,r=1}^{Q} \mathcal{N}_K\left(Y_{qr}; \mu_{qr}^{\nu}(\boldsymbol{\tau}), \mathrm{diag}((\sigma_{qr}^{\nu})^2(\boldsymbol{\tau}))\right),$$

- $W_{qr} = \{W_{ij} : C_{iq}C_{jr} = 1\}$, the set of documents sent from cluster q to cluster r.
- $\tilde{W}_{qr} = \mathbb{E}_R\left[W_{qr}\right] = \sum_{i \neq j} \tau_{iq}\tau_{jr}A_{ij}W_{ij}$.

.

Inference: meta-documents construction

Regarding the latent topic proportions θ_{qr}, we propose a new encoding of textual data:

$$R(\mathbf{Y}) = \prod_{q,r=1}^{Q} R(Y_{qr}) = \prod_{q,r=1}^{Q} \mathcal{N}_K\left(Y_{qr}; \mu_{qr}^{\nu}(\boldsymbol{\tau}), \mathrm{diag}((\sigma_{qr}^{\nu})^2(\boldsymbol{\tau}))\right),$$

- $W_{qr} = \{W_{ij} : C_{iq}C_{jr} = 1\}$, the set of documents sent from cluster q to cluster r.
- $\tilde{W}_{qr} = \mathbb{E}_R\left[W_{qr}\right] = \sum_{i \neq j} \tau_{iq}\tau_{jr}A_{ij}W_{ij}$.
- $\mu_{qr}^{\nu}(\boldsymbol{\tau}) = f(\tilde{W}_{qr}; \nu_m)$ and $\sigma_{qr}^{\nu}(\boldsymbol{\tau}) = f(\tilde{W}_{qr}; \nu_v)$.

Inference: meta-documents construction

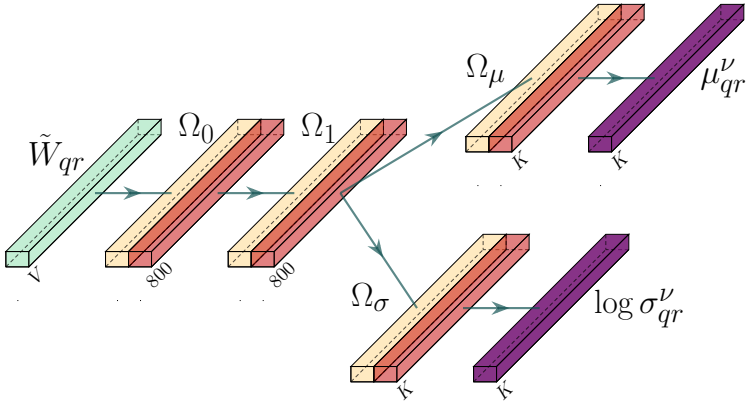


Figure 5: The deep neural network architecture of the meta-document encoder, where $\nu = \{\Omega_0, \Omega_1, \Omega_\mu, \Omega_\sigma\}$.

ELBO derivation

Thanks to the previous assumptions, the ELBO can be computed explicitly as follow:

$$
\begin{aligned}
\mathscr{L}(R(\cdot); \boldsymbol{\alpha}, \boldsymbol{\rho}) = & \mathbb{E}_R \left[\log p(\mathbf{W} \mid \mathbf{C}, \mathbf{A}, \mathbf{Y}, \boldsymbol{\alpha}, \boldsymbol{\rho})\right] \\
& + \mathbb{E}_R \left[\log p(\mathbf{Y})\right] - \mathbb{E}_R \left[\log R(\mathbf{Y})\right] \\
& + \mathbb{E}_R \left[\log p(\mathbf{A} \mid \mathbf{C}, \boldsymbol{\Pi})\right] \\
& + \mathbb{E}_R \left[\log p(\mathbf{C} \mid \gamma)\right] - \mathbb{E}_R \left[\log R(\mathbf{C})\right] \\
& + \mathbb{E}_R \left[\log p(\boldsymbol{\Pi})\right] - \mathbb{E}_R \left[\log R(\boldsymbol{\Pi})\right] \\
& + \mathbb{E}_R \left[\log p(\gamma)\right] - \mathbb{E}_R [\log R(\gamma)].
\end{aligned}
$$

Optimisation: first order conditions

The variational parameters $\tilde{\pi}$ and $\tilde{\gamma}$ only depend on $\boldsymbol{\tau}$ and are updated as follow[4]:

$$\tilde{\gamma_q} = \gamma_{0q} + \sum_{i=1}^{M} \tau_{iq},$$

$$\tilde{\pi}_{qr1} = \pi^0_{qr1} + \sum_{i \neq j}^{M} \tau_{iq}\tau_{jr}A_{ij}, \quad \tilde{\pi}_{qr2} = \pi^0_{qr2} + \sum_{i \neq j}^{M} \tau_{iq}\tau_{jr}(1 - A_{ij}).$$

[4] Latouche, Birmele, and Ambroise (2012).

Optimisation: first order conditions

The variational parameters $\tilde{\pi}$ and $\tilde{\gamma}$ only depend on $\boldsymbol{\tau}$ and are updated as follow[4]:

$$\tilde{\gamma_q} = \gamma_{0q} + \sum_{i=1}^{M} \tau_{iq},$$

$$\tilde{\pi}_{qr1} = \pi^0_{qr1} + \sum_{i \neq j}^{M} \tau_{iq}\tau_{jr}A_{ij}, \quad \tilde{\pi}_{qr2} = \pi^0_{qr2} + \sum_{i \neq j}^{M} \tau_{iq}\tau_{jr}(1 - A_{ij}).$$

To perform stochastic gradient descent, $\tau_i \in \Delta_Q$ is mapped to an unconstrained $\xi_i \in \mathbb{R}^{Q-1}$, through the following transformation:

$$\xi_{iq} = \ln(\tau_{iq}) - \ln(\tau_{iQ}), \ \forall q \in \{1, \ldots, Q-1\}.$$

[4] Latouche, Birmele, and Ambroise (2012).

Optimisation: first order conditions

The variational parameters $\tilde{\pi}$ and $\tilde{\gamma}$ only depend on $\boldsymbol{\tau}$ and are updated as follow[4]:

$$\tilde{\gamma_q} = \gamma_{0q} + \sum_{i=1}^{M} \tau_{iq},$$

$$\tilde{\pi}_{qr1} = \pi^0_{qr1} + \sum_{i \neq j}^{M} \tau_{iq}\tau_{jr}A_{ij}, \quad \tilde{\pi}_{qr2} = \pi^0_{qr2} + \sum_{i \neq j}^{M} \tau_{iq}\tau_{jr}(1 - A_{ij}).$$

To perform stochastic gradient descent, $\tau_i \in \Delta_Q$ is mapped to an unconstrained $\xi_i \in \mathbb{R}^{Q-1}$, through the following transformation:

$$\xi_{iq} = \ln(\tau_{iq}) - \ln(\tau_{iQ}), \ \forall q \in \{1, \ldots, Q-1\}.$$

Then, $\boldsymbol{\xi}$ as well as $\boldsymbol{\rho}$ and $\boldsymbol{\alpha}$ are directly optimised by a stochastic gradient descent algorithm. However, $\boldsymbol{\nu}$ requires an additional step: the reparametrisation trick.

[4] Latouche, Birmele, and Ambroise (2012).

The reparametrisation trick[5]

How to compute the gradient $\frac{\partial}{\partial \nu}\mathscr{L}(R(\cdot); \boldsymbol{\alpha}, \boldsymbol{\rho})$?

$$\frac{\partial}{\partial \nu}\mathscr{L}(R(\cdot); \boldsymbol{\alpha}, \boldsymbol{\rho}) = \frac{\partial}{\partial \nu}\mathbb{E}_R\left[\log p(\mathbf{W} \mid \mathbf{C}, \mathbf{A}, \mathbf{Y}, \boldsymbol{\alpha}, \boldsymbol{\rho})\right] - \frac{\partial}{\partial \nu}\overbrace{\mathrm{KL}(R(\mathbf{Y}) \mid p(\mathbf{Y}))}^{\text{analytical form}}.$$

Since $R(\cdot)$ depends on ν, we cannot interchange the derivative and the integral in the term on the left-hand side.

The reparametrisation trick removes this dependency by sampling $\epsilon \sim \mathcal{N}_K(0, \mathbf{I}_K)$ and taking $Y_{qr} = \mu^{\nu}_{qr}(\tau) + \sigma^{\nu}_{qr}(\tau)\epsilon$, such that the following holds:

$$\begin{aligned}\frac{\partial}{\partial \nu}\mathbb{E}_R\left[\log p(\mathbf{W} \mid \mathbf{C}, \mathbf{A}, \mathbf{Y}, \boldsymbol{\alpha}, \boldsymbol{\rho})\right] &= \frac{\partial}{\partial \nu}\mathbb{E}_\epsilon\Big[\mathbb{E}_C\left[\log p(\mathbf{W} \mid \mathbf{C}, \mathbf{A}, \mathbf{Y}, \boldsymbol{\alpha}, \boldsymbol{\rho})\right]\Big] \\ &= \mathbb{E}_\epsilon\Big[\frac{\partial}{\partial \nu}\mathbb{E}_C\left[\log p(\mathbf{W} \mid \mathbf{C}, \mathbf{A}, \mathbf{Y}, \boldsymbol{\alpha}, \boldsymbol{\rho})\right]\Big].\end{aligned}$$

A Monte-Carlo estimate of this last expression can now be computed.

[5] Kingma and Welling (2014); Rezende, Mohamed, and Wierstra (2014).

Numerical experiments

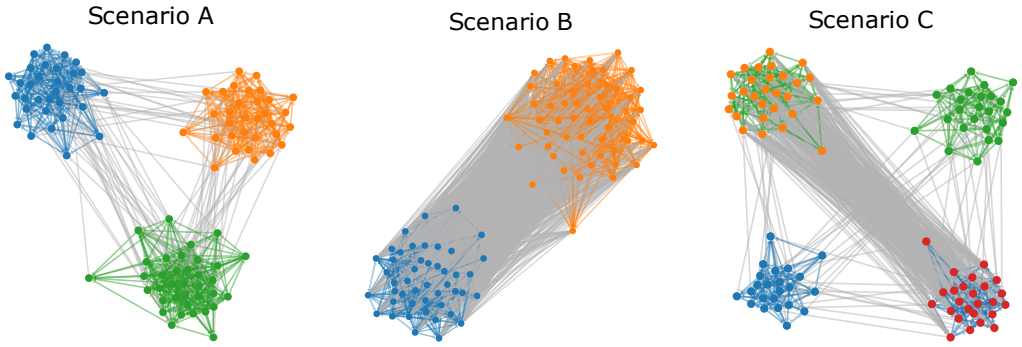


Figure 6: An example of each scenario is presented. The node colours denote the cluster memberships and the edge colours denote the most-used topic within the corresponding documents. The scenarios A, B and C are composed of 3, 1 and 3 communities respectively.

Evaluation of the ELBO as a model selection criterion

K \ Q	Scenario A					Scenario B					Scenario C				
	2	**3**	4	5	10	**2**	3	4	5	10	2	3	**4**	5	10
2	0	**94**	6	0	0	**74**	24	2	0	0	0	0	**92**	8	0
3	0	**90**	10	0	0	**78**	18	4	0	0	0	0	**90**	10	0
4	0	**78**	20	2	0	**76**	20	4	0	0	0	0	**94**	6	0
5	0	**86**	14	0	0	**68**	28	4	0	0	0	0	**84**	16	0
10	0	**88**	10	2	0	**82**	18	0	0	0	0	0	**86**	14	0

Table 1: This table presents the percentage of time a number of clusters have been selected on 50 simulated networks. The experiment is repeated for different values of K, and Scenarios A, B and C. For instance, in Scenario A with $K = 3$, the model with $Q = 3$ clusters was selected in 90% of cases.

	Scenario A		Scenario B		Scenario C	
	Node ARI	Edge ARI	Node ARI	Edge ARI	Node ARI	Edge ARI
ETSBM	$\mathbf{0.98 \pm 0.06}$	$\mathbf{0.83 \pm 0.07}$	$\mathbf{1.00 \pm 0.00}$	0.86 ± 0.03	$\mathbf{0.91 \pm 0.12}$	$\mathbf{0.84 \pm 0.12}$
STBM	0.75 ± 0.27	0.82 ± 0.22	$\mathbf{1.00 \pm 0.00}$	$\mathbf{1.00 \pm 0.00}$	0.63 ± 0.19	0.77 ± 0.15
SBM	0.96 ± 0.05	———	0.00 ± 0.00	———	0.63 ± 0.11	———
SC	0.98 ± 0.08	———	0.00 ± 0.00	———	0.60 ± 0.11	———
LDA	———	0.77 ± 0.09	———	0.88 ± 0.02	———	0.84 ± 0.04
ETM	———	0.83 ± 0.08	———	0.85 ± 0.03	———	0.86 ± 0.04

Table 2: Benchmark of our model against STBM, SBM, SC and LDA. The results are obtained by taking the best out of 10 initialisations, and averaged over 50 graphs.

Analysis of the Tweets published or "retweeted" prior to the 2022 French presidential election

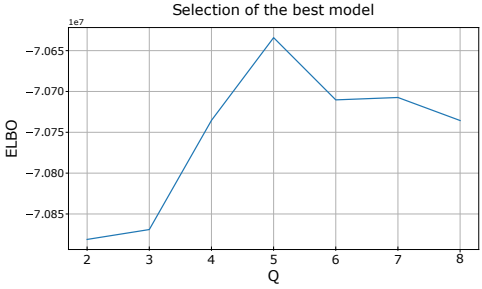


Figure 7: After running ETSBM with a different number of clusters Q, the ELBO suggests keeping five clusters.

Analysis of the Tweets published or "retweeted" prior to the 2022 French presidential election

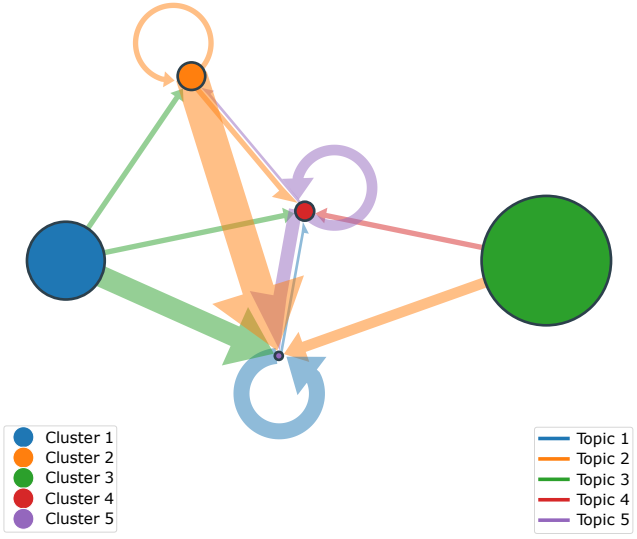


Figure 8: ETSBM clusters of twitter accounts and topics of the tweets.

Analysis of the Tweets published or "retweeted" prior to the 2022 French presidential election

Topics

tour	heure	macron	melenchon	zemmour
tout	melenchonvagagner	candidat	jadot	eric
faire	monde	emmanuel	jlm	jevotezemmour
aller	erepublique	campagne	roussel	soutenir
voter	melenchon	zemmour	voter	jevotezemmourle

Figure 9: The most important words according to ETSBM topics.

Objective of this thesis

Main objective of this thesis: develop new methodologies to improve node clustering and topic modelling on a network with textual edges to improve the relevancy of the results.

- Can advancements in neural topic models help to improve clustering ?

→ First, we propose the embedded topics in the stochastic block model.

- Can deep encoding of the documents and of the connections help to solve node clustering as well as topic modelling ?
- How to estimate node positions for visualisation purposes in an end-to-end methodology ?

→ Second, we present the deep latent position topic model.

Finally, we discard the textual edges to focus on the connectivity.

- Can block modelling and position estimation be performed simultaneously, based on a deep probabilistic model ?

→ The deep latent positional block model is introduced in the third section.

Objective of this thesis

Main objective of this thesis: develop new methodologies to improve node clustering and topic modelling on a network with textual edges to improve the relevancy of the results.

- Can advancements in neural topic models help to improve clustering ?

→ First, we propose the embedded topics in the stochastic block model.

- Can deep encoding of the documents and of the connections help to solve node clustering as well as topic modelling ?
- How to estimate node positions for visualisation purposes in an end-to-end methodology ?

→ Second, we present the deep latent position topic model.

Finally, we discard the textual edges to focus on the connectivity.

- Can block modelling and position estimation be performed simultaneously, based on a deep probabilistic model ?

→ The deep latent positional block model is introduced in the third section.

Deep latent position topic model

1. Embedded topics in the stochastic block model
2. Deep latent position topic model
3. Deep latent position block model
4. Conclusion and perspectives

Generative model: Assumptions made by Deep-LPTM about the generation of networks with textual edges

- Node cluster memberships: $C_i \overset{\text{i.i.d}}{\sim} \mathcal{M}_Q(1, \gamma)$.
- Node embedding: $Z_i \mid C_{iq} = 1 \sim \mathcal{N}_p(\mu_q, \sigma_q^2 \mathbf{I}_p)$.
- Edge from node i to j, $\eta_{ij} = \kappa - \|Z_i - Z_j\|$:

$$A_{ij} \mid \{Z_i, Z_j\} \sim \mathcal{B}((1 + e^{-\eta_{ij}})^{-1}).$$

- Edge embedding:

$$Y_{ij} \mid \{A_{ij} C_{iq} C_{jr} = 1\} \sim \mathcal{N}_K(m_{qr}, s_{qr}^2 \mathbf{I}_K).$$

- Word within documents sent from i to j

$$W_{ij}^{dn} \mid \{A_{ij} = 1, \theta_{ij} = \text{softmax}(Y_{ij})\} \sim \mathcal{M}_V\left(1, \boldsymbol{\beta}^\top \theta_{ij}\right).$$

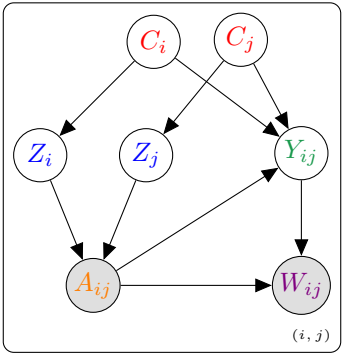


Figure 10: Graphical representation of Deep-LPTM.

Marginal likelihood

Denoting Θ the set of all model parameters,

$$\log p(\mathbf{A}, \mathbf{W} \mid \Theta) = \log \left(\sum_{\mathbf{C}} \int_{\mathbf{Z}} \int_{\mathbf{Y}} p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{Z}, \mathbf{Y} \mid \Theta) d\mathbf{Z} d\mathbf{Y} \right). \tag{1}$$

Marginal likelihood

Denoting Θ the set of all model parameters,

$$\log p(\mathbf{A}, \mathbf{W} \mid \Theta) = \log \left(\sum_{\mathbf{C}} \int_{\mathbf{Z}} \int_{\mathbf{Y}} p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{Z}, \mathbf{Y} \mid \Theta) d\mathbf{Z} d\mathbf{Y} \right). \tag{1}$$

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.

Marginal likelihood

Denoting Θ the set of all model parameters,

$$\log p(\mathbf{A}, \mathbf{W} \mid \Theta) = \log \left(\sum_{\mathbf{C}} \int_{\mathbf{Z}} \int_{\mathbf{Y}} p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{Z}, \mathbf{Y} \mid \Theta) d\mathbf{Z} d\mathbf{Y} \right). \tag{1}$$

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

Assumptions regarding the variational distributions:

Inference

Assumptions regarding the variational distributions:

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

Inference

Assumptions regarding the variational distributions:

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

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

Assumptions regarding the variational distributions:

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

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

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

Assumptions regarding the variational distributions:

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

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

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

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

where $\left(\mu_{\phi_Z}, \sigma^2_{\phi_Z}\right)$ are the outputs of the encoder of a variational graph auto encoder [6] and $\left(\mu_{\phi_Y}, \sigma^2_{\phi_Y}\right)$ the outputs of ETM encoder[7].

[6] Thomas N Kipf and Welling (2016)
[7] Dieng, Ruiz, and Blei (2020)

Proposition

Let $R(\cdot)$ be a variational distribution complying with the previous assumptions. The parameters of the node embedding distributions maximising the ELBO are given by:

$$\mu_q = \frac{1}{N_q} \sum_{i=1}^{N} \tau_{iq} \mu_{\phi_Z}(\mathbf{A})_i, \tag{2}$$

$$\sigma_q^2 = \frac{1}{pN_q} \sum_{i=1}^{N} \tau_{iq} \left(\|\mu_{\phi_Z}(\mathbf{A})_i - \mu_q\|_2^2 + p\sigma_{\phi_Z}^2(\mathbf{A})_i \right), \tag{3}$$

where $N_q = \sum_{i=1}^{N} \tau_{iq}$ is the posterior mean of the number of nodes in cluster q.

Proposition

Let $R(\cdot)$ be a variational distribution complying with the already mentioned assumptions, the parameters of the edge embedding distributions maximising the ELBO are given by:

$$m_{qr} = \frac{1}{N_{qr}} \sum_{i,j=1}^{N} A_{ij} \tau_{iq} \tau_{jr} \mu_{\phi_Y}(W_{ij}), \tag{4}$$

$$s^2_{qr} = \frac{1}{K N_{qr}} \sum_{i,j=1}^{N} A_{ij} \tau_{iq} \tau_{jr} \left[\|\mu_{\phi_Y}(W_{ij}) - m_{qr}\|_2^2 + \sum_{k=1}^{K} \sigma^2_{\phi_Y}(W_{ij})_k \right], \tag{5}$$

where $N_{qr} = \sum_{i,j=1}^{N} A_{ij} \tau_{iq} \tau_{jr}$ denotes the expected number of documents sent from cluster q to cluster r under the approximated posterior distribution.

Optimisation: the clusters encapsulate both information

Proposition

Let $R(\cdot)$ be a variational distribution complying with the already mentioned assumptions, the parameter τ_{iq} maximising the ELBO is given by:

$$\tau_{iq} \propto \gamma_q \exp\Big\{ -\mathrm{KL}_q^{Z_i} - \sum_{j \neq i} \sum_{r=1}^{Q} \Big(A_{ij} \tau_{jr} \, \mathrm{KL}_{qr}^{Y_{ij}} + A_{ji} \tau_{jr} \, \mathrm{KL}_{rq}^{Y_{ji}} \Big) \Big\},$$

where

$$\mathrm{KL}_q^{Z_i} = \mathrm{KL}\Big(\overbrace{\mathcal{N}_p(\mu_{\phi_Z}(\mathbf{A})_i, \sigma^2_{\phi_Z}(\mathbf{A})_i \mathbf{I}_p)}^{\text{variational distribution of node embedding}} \;\Big\|\; \overbrace{\mathcal{N}_p\left(\mu_q, \sigma_q^2 \mathbf{I}_p\right)}^{\text{distribution of cluster } q \text{ embedding}} \Big),$$

$$\mathrm{KL}_{qr}^{Y_{ij}} = \mathrm{KL}\Big(\underbrace{\mathcal{N}_K\left(\mu_{\phi_Y}(W_{ij}), \mathrm{diag}\left(\sigma^2_{\phi_Y}(W_{ij})\right)\right)}_{\text{variational distribution of edge embedding}} \;\Big\|\; \underbrace{\mathcal{N}_K(m_{qr}, s^2_{qr}\mathbf{I}_K)}_{\substack{\text{distribution of document} \\ \text{embedding sent from cluster } q \text{ to } r}} \Big).$$

Optimisation of the encoders

The parameters of the graph convolutional network encoder as well as the parameters of the encoder of the neural topic model are optimised using a MC estimate of the gradient obtained thanks to the reparametrisation trick, and using a stochastic gradient descent algorithm.

Optimisation of the encoders

The parameters of the graph convolutional network encoder as well as the parameters of the encoder of the neural topic model are optimised using a MC estimate of the gradient obtained thanks to the reparametrisation trick, and using a stochastic gradient descent algorithm. What does that mean in practice ?

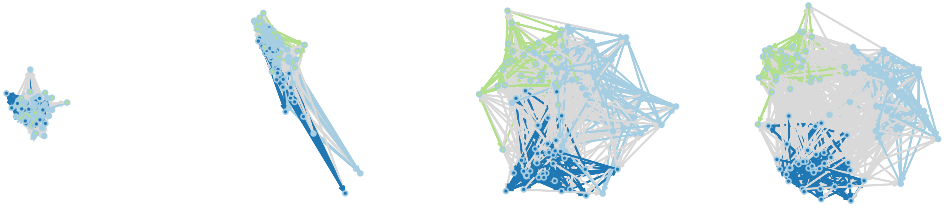


Figure 11: Example of the node positions evolutions on Scenario A during the training.

Model selection

How to select P, Q, and K ? We propose to use the joint distribution[8]:

$$\log p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{Z}, \mathbf{Y} \mid \mathcal{M}, Q, K, P) = \log \int_{\theta} p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{Z}, \mathbf{Y} \mid \theta, \mathcal{M}, Q, K, P) p(\theta) d\theta.$$

[8] Biernacki, Celeux, and Govaert (2000)

Model selection

How to select P, Q, and K ? We propose to use the joint distribution[8]:

$$\log p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{Z}, \mathbf{Y} \mid \mathcal{M}, Q, K, P) = \log \int_{\theta} p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{Z}, \mathbf{Y} \mid \theta, \mathcal{M}, Q, K, P) p(\theta) d\theta.$$

$\longrightarrow$ this quantity is intractable. Therefore, we estimate it using a BIC-like approximation.

[8] Biernacki, Celeux, and Govaert (2000)

How to select P, Q, and K ? We propose to use the joint distribution[8]:

$$\log p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{Z}, \mathbf{Y} \mid \mathcal{M}, Q, K, P) = \log \int_{\theta} p(\mathbf{A}, \mathbf{W}, \mathbf{C}, \mathbf{Z}, \mathbf{Y} \mid \theta, \mathcal{M}, Q, K, P) p(\theta) d\theta.$$

$\longrightarrow$ this quantity is intractable. Therefore, we estimate it using a BIC-like approximation.

Proposed estimate:

$$ICL(\mathcal{M}, Q, K, P, \hat{\mathbf{C}}, \hat{\mathbf{Z}}, \hat{\mathbf{Y}}) = \max_{\theta} \log p(\mathbf{A}, \mathbf{W}, \hat{\mathbf{C}}, \hat{\mathbf{Z}}, \hat{\mathbf{Y}} \mid \theta, \mathcal{M}, Q, K, P) - \Omega(\mathcal{M}, Q, K, P),$$

with $\hat{\mathbf{C}}$ $\hat{\mathbf{Z}}$ and $\hat{\mathbf{Y}}$ the **maximum-a-posteriori** estimates, and $\Omega(\mathcal{M}, Q, K, P)$ the **penalty** from BIC-like approximations.

[8] Biernacki, Celeux, and Govaert (2000)

Assessment of IC2L criterion

	Scenario A $Q^\star = 3$	Scenario B $Q^\star = 2$	Scenario C $Q^\star = 4$
$Q = 2$	0	10	0
$Q = 3$	10	0	0
$Q = 4$	0	0	10
$Q = 5$	0	0	0
$Q = 10$	0	0	0

Table 3: Number of times a value Q is selected by the IC2L criterion over 10 graphs with the true value of K and $P = 2$.

Benchmark

	Scenario A	Scenario B	Scenario C
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	$\mathbf{0.89 \pm 0.15}$
Deep-LPTM - PT	$\mathbf{1.00 \pm 0.01}$	1.00 ± 0.00	0.85 ± 0.18

Table 4: ARI of the node 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.

Analysis of ENRON emails network: Entire network visualisation

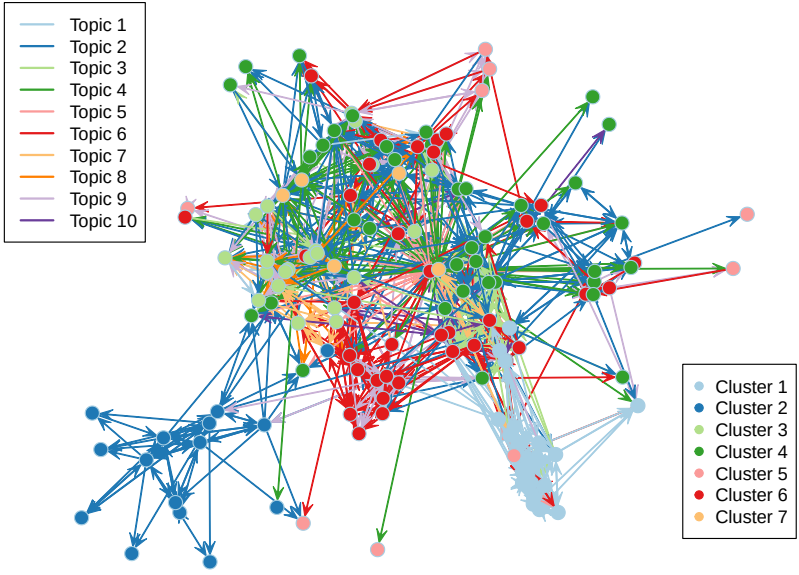


Figure 12: ENRON network representation estimated with Deep-LPTM

Analysis of ENRON emails network: meta network

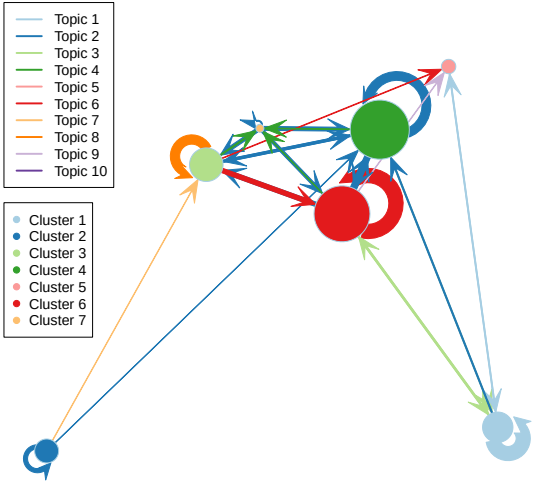


Figure 13: Networks plotted using Deep-LPTM estimates for the nodes positions as well as the node clusters and the edge topics (represented by their colour)

Objective of this thesis

Main objective of this thesis: develop new methodologies to improve node clustering and topic modelling on a network with textual edges to improve the relevancy of the results.

- Can advancements in neural topic models help to improve clustering ?

→ First, we propose the embedded topics in the stochastic block model.

- Can deep encoding of the documents and of the connections help to solve node clustering as well as topic modelling ?
- How to estimate node positions for visualisation purposes in an end-to-end methodology ?

→ Second, we present the deep latent position topic model.

Finally, we discard the textual edges to focus on the connectivity.

- Can block modelling and position estimation be performed simultaneously, based on a deep probabilistic model ?

→ The deep latent positional block model is introduced in the third section.

Objective of this thesis

Main objective of this thesis: develop new methodologies to improve node clustering and topic modelling on a network with textual edges to improve the relevancy of the results.

- Can advancements in neural topic models help to improve clustering ?

→ First, we propose the embedded topics in the stochastic block model.

- Can deep encoding of the documents and of the connections help to solve node clustering as well as topic modelling ?
- How to estimate node positions for visualisation purposes in an end-to-end methodology ?

→ Second, we present the deep latent position topic model.

Finally, we discard the textual edges to focus on the connectivity.

- Can block modelling and position estimation be performed simultaneously, based on a deep probabilistic model ?

→ The deep latent positional block model is introduced in the third section.

Deep latent position block model

1. Embedded topics in the stochastic block model
2. Deep latent position topic model
3. Deep latent position block model
4. Conclusion and perspectives

The latent position cluster model

- In this work, we only consider networks without textual edges.
- The goal is to provide a block model approach that also provides node positions for visualisation purposes.
- We also aim at analysing disassortative graph, or stars, which cannot be done with positional methodologies.

The latent position model [9]

- Node embedding: $Z_i, Z_j \in \mathbb{R}^p$,
- Edge from node i to j, $\eta_{ij} = \kappa - \|Z_i - Z_j\|$:

$$A_{ij} \mid \{Z_i, Z_j\} \sim \mathcal{B}((1 + e^{-\eta_{ij}})^{-1}).$$

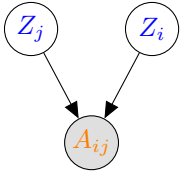


Figure 14: Graphical representation of LPM .

[9] Hoff, Raftery, and Handcock (2002)

The latent position cluster model [9]

- Node cluster memberships: $C_i \overset{\text{i.i.d}}{\sim} \mathcal{M}_Q(1, \gamma)$,
- $Z_i \mid C_{iq} = 1 \sim \mathcal{N}_p\left(\mu_q, \sigma_q^2 \mathbf{I}_p\right)$,
- Edge from node i to j, $\eta_{ij} = \kappa - \|Z_i - Z_j\|$:

$$A_{ij} \mid \{Z_i, Z_j\} \sim \mathcal{B}((1 + e^{-\eta_{ij}})^{-1}).$$

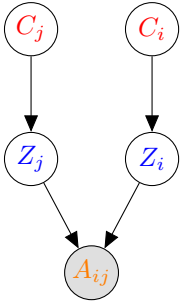


Figure 14: Graphical representation of LPCM.

[9] Handcock, Raftery, and Tantrum (2007)

The Deep-latent position model[11]

Relying on variational autoencoders to impose a structure on the latent space.

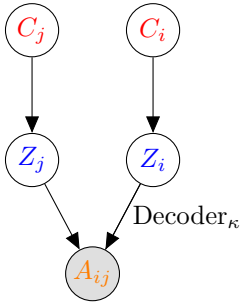


Figure 15: Graphical representation of Deep-LPM.

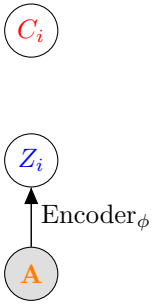


Figure 16: Structure of the variational distribution.

where

$$\text{Decoder}_\kappa(Z_i, Z_j) = (1 + e^{-\eta_{ij}})^{-1}, \text{ as in the LPCM,}$$
$$\text{Encoder}_\phi(\mathbf{A}) = \text{GCN}(\mathbf{A})_i = (\mu_\phi(\mathbf{A})_i, \sigma_\phi(\mathbf{A})_i)^{[10]}.$$

[10] Thomas N. Kipf and Welling (2017)
[11] Liang et al. (2022)

The Deep-Latent position block model

To capture any type of connectivity pattern within the network, we propose a new decoder incorporating a block modelling approach.

The Deep-Latent position block model

To capture any type of connectivity pattern within the network, we propose a new decoder incorporating a block modelling approach.

- Node cluster memberships: $C_i \mid \gamma \overset{\text{i.i.d}}{\sim} \mathcal{M}_Q(1, \gamma)$,
- Node embedding: $Z_i \mid C_{iq} = 1 \sim \mathcal{N}_{Q-1}\left(\mu_q, \sigma_q^2 \mathbf{I}_{Q-1}\right)$,

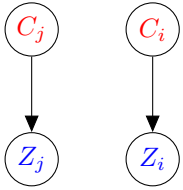


Figure 17: Graphical representation of Deep-LPBM.

The Deep-Latent position block model

To capture any type of connectivity pattern within the network, we propose a new decoder incorporating a block modelling approach.

- Node cluster memberships: $C_i \mid \gamma \overset{\text{i.i.d}}{\sim} \mathcal{M}_Q(1, \gamma)$,
- Node embedding: $Z_i \mid C_{iq} = 1 \sim \mathcal{N}_{Q-1}\left(\mu_q, \sigma_q^2 \mathbf{I}_{Q-1}\right)$,
- Edge from node i to j:

$$A_{ij} \mid \{Z_i, Z_j, \mathbf{\Pi}\} \sim \mathcal{B}(\eta_i^\top \mathbf{\Pi} \eta_j),$$

where $\mathbf{\Pi} \in \mathcal{M}_{Q \times Q}((0,1))$ and $\eta_i = \text{softmax}(Z_i)$.

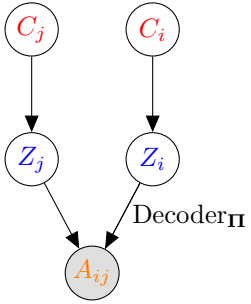


Figure 17: Graphical representation of Deep-LPBM.

Inference: log-likelihood

Denoting, $\Theta = \{\mathbf{\Pi}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\gamma}\}$, the marginal log-likelihood is given by:

$$\mathcal{L}(\Theta; \mathbf{A}) = \log p(\mathbf{A} \mid \Theta) = \log \left(\sum_{\mathbf{C}} \int_{\mathbf{Z}} p(\mathbf{A}, \mathbf{C}, \mathbf{Z} \mid \Theta) d\mathbf{Z} \right). \tag{6}$$

For the same reasons as before, we use variational approximations of the posterior to estimate the parameters

Inference: variational assumptions

We rely on a mean-field variational inference and specify the distribution of the latent variables below:

$$R(\mathbf{C}, \mathbf{Z} \mid \mathbf{A}) = R(\mathbf{C}) R(\mathbf{Z} \mid \mathbf{A}), \tag{7}$$

$$R(\mathbf{C}) = \prod_{i=1}^{N} R_{\tau_i}(C_i) = \prod_{i=1}^{N} \mathcal{M}_Q(C_i; 1, \tau_i), \tag{8}$$

$$R(\mathbf{Z} \mid \mathbf{A}) = \prod_{i=1}^{N} R_\phi(Z_i \mid \mathbf{A}) = \prod_{i=1}^{N} \mathcal{N}_{Q-1}(Z_i; \mu_\phi(\mathbf{A})_i, \sigma^2_\phi(\mathbf{A})_i \mathbf{I}_{Q-1}). \tag{9}$$

Some details about VGAE[12]

Denoting $\tilde{\mathbf{A}} = \mathbf{D}^{-1/2}(\mathbf{A} + \mathbf{I}_N)\mathbf{D}^{-1/2}$, the graph convolutional network can be summarised as

[12] Thomas N Kipf and Welling (2016).

Some details about VGAE[12]

Denoting $\tilde{\mathbf{A}} = \mathbf{D}^{-1/2}(\mathbf{A} + \mathbf{I}_N)\mathbf{D}^{-1/2}$, the graph convolutional network can be summarised as

$$\mu_\phi(\mathbf{A}) = \tilde{\mathbf{A}}\,\mathrm{ReLU}(\tilde{\mathbf{A}}\mathbf{\Omega}_0)\mathbf{\Omega}_\mu,$$
$$\log\sigma^2_\phi(\mathbf{A}) = \tilde{\mathbf{A}}\,\mathrm{ReLU}(\tilde{\mathbf{A}}\mathbf{\Omega}_0)\mathbf{\Omega}_\sigma,$$

where

- $\mathrm{ReLU}(x) = (\max(0, x_1), \ldots, \max(0, x_F))$ if $x \in \mathbb{R}^F$,
- $\mathbf{\Omega}_0 \in \mathcal{M}_{N\times D}(\mathbb{R})$ with $D = 64$ in all the experiments we carried out,
- $\mathbf{\Omega}_\mu, \mathbf{\Omega}_\sigma \in \mathcal{M}_{D\times(Q-1)}(\mathbb{R})$.

[12] Thomas N Kipf and Welling (2016).

Inference: derivation of the ELBO

$$
\begin{aligned}
\mathscr{L}(R(\cdot);\Theta) =& \mathbb{E}_R\left[\log \frac{p(\mathbf{A},\mathbf{C},\mathbf{Z})}{R(\mathbf{C},\mathbf{Z})}\right] \\
=& \mathbb{E}_R\left[\log p(\mathbf{A} \mid \mathbf{Z},\mathbf{\Pi})\right] \\
&+ \mathbb{E}_R\left[\log p(\mathbf{Z} \mid \mathbf{C},\boldsymbol{\mu},\boldsymbol{\sigma})\right] - \mathbb{E}_R\left[\log R(\mathbf{Z} \mid \mathbf{A})\right] \\
&+ \mathbb{E}_R\left[\log p(\mathbf{C} \mid \boldsymbol{\gamma})\right] - \mathbb{E}_R\left[\log R(\mathbf{C})\right]. \\
=& \underbrace{\mathbb{E}_R\left[\log p(\mathbf{A} \mid \mathbf{Z},\mathbf{\Pi})\right]}_{\text{Reconstruction loss}} \underbrace{- \mathrm{KL}(R(\mathbf{Z} \mid \mathbf{A})||p(\mathbf{Z} \mid \mathbf{C},\boldsymbol{\mu},\boldsymbol{\sigma})) - \mathrm{KL}(R(\mathbf{C})||p(\mathbf{C} \mid \boldsymbol{\gamma}))}_{\text{Regularising term}}.
\end{aligned}
$$

Inference: derivation of the ELBO

$$
\begin{aligned}
\mathscr{L}(R(\cdot);\Theta) = & \sum_{i,j=1}^{N} \left\{ A_{ij} \mathbb{E}_R \left[\log \eta_i^\top \mathbf{\Pi} \eta_j \right] + (1 - A_{ij}) \mathbb{E}_R \left[\log \left(1 - \eta_i^\top \mathbf{\Pi} \eta_j \right) \right] \right\} \\
& - \sum_{i=1}^{N} \sum_{q=1}^{Q} \tau_{iq} \, \mathrm{KL}_{iq} \left(\mu_\phi(\mathbf{A})_i, \sigma_\phi(\mathbf{A})_i, \mu_q, \sigma_q \right) \\
& - \sum_{i=1}^{N} \sum_{q=1}^{Q} \tau_{iq} \log \frac{\tau_{iq}}{\gamma_q},
\end{aligned}
\tag{10}
$$

where

$$
\begin{aligned}
& \mathrm{KL}_{iq} \left(\mu_\phi(\mathbf{A})_i, \sigma_\phi(\mathbf{A})_i, \mu_q, \sigma_q \right) \\
& = \mathrm{KL} \Big(\underbrace{\mathcal{N}_{Q-1}(\mu_\phi(\mathbf{A})_i, \sigma_\phi^2(\mathbf{A})_i \mathbf{I}_{Q-1})}_{\text{variational distribution of } Z_i} \; || \; \underbrace{\mathcal{N}_{Q-1} \left(\mu_q, \sigma_q^2 \mathbf{I}_{Q-1} \right)}_{\substack{\text{embedding distribution of} \\ \text{a node from cluster } q}} \Big).
\end{aligned}
$$

Optimisation: update of the posterior probabilities

Proposition

Let $\mathscr{L}(R(\cdot);\Theta)$ denote the ELBO, the first-order conditions with respect to $\boldsymbol{\tau} = (\tau_i)_i$ give the following updates for any node i and cluster q

$$\tau_{iq} = \frac{\gamma_q e^{-\mathrm{KL}_{iq}}}{\sum_{r=1}^{Q} \gamma_r e^{-\mathrm{KL}_{ir}}}. \tag{11}$$

Optimisation: update of the node embedding distribution and the posterior cluster proportions

Proposition

Let $\mathscr{L}(R(\cdot);\Theta)$ denote the ELBO, the first-order conditions with respect to γ, $(\mu_q)_q$ and $(\sigma_q)_q$ give the following updates:

$$\gamma_q = \frac{1}{N}\sum_{i=1}^{N}\tau_{iq}, \tag{12}$$

$$\mu_q = \left(\sum_{i=1}^{N}\tau_{iq}\right)^{-1}\sum_{i=1}^{N}\tau_{iq}\mu_\phi(\mathbf{A})_i, \tag{13}$$

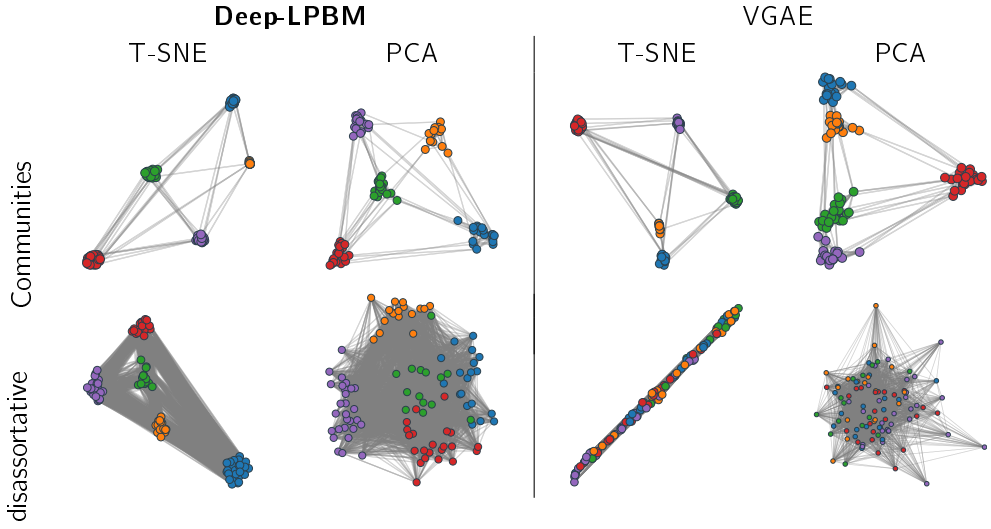
$$\sigma_q^2 = \left((Q-1)\sum_{i=1}^{N}\tau_{iq}\right)^{-1}\sum_{i=1}^{N}\tau_{iq}\Big(\|\mu_\phi(\mathbf{A})_i - \mu_q\|_2^2 + (Q-1)\sigma_\phi^2(\mathbf{A})_i\Big). \tag{14}$$

Promising results on synthetic networks

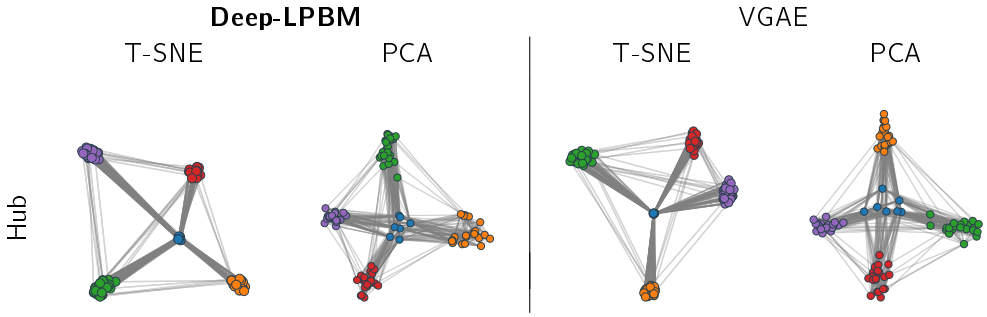
	Communities	disassortative	Hub
VGAE + Kmeans	1.00 ± 0.01	0.02 ± 0.02	0.97 ± 0.06
Deep LPM	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00
Deep LPBM	$\mathbf{1.00 \pm 0.00}$	$\mathbf{0.80 \pm 0.08}$	$\mathbf{1.00 \pm 0.00}$

Table 5: ARI of the partition obtained by a K-means algorithm applied on the VGAE node embeddings, the Deep-LPM and Deep-LPBM partitions. The mean and standard deviations were obtained over 10 networks for each graph structure.

Visualisation results on three different network structures



Results on three different network structures



Conclusion and perspectives

1. Embedded topics in the stochastic block model
2. Deep latent position topic model
3. Deep latent position block model
4. Conclusion and perspectives

Conclusion and perspectives

What we can affirm after these three contributions:

- Advancements in neural topic model helped to improve node clustering results.
- Graph neural network can be efficiently coupled with a neural topic model to provide an intelligible visualisation as well as an efficient node clustering methodology.
- Using a graph convolutional network to encode the node connections into node positions enables to simultaneously estimate a block model as well as a positional model.

Conclusion and perspectives

What we can affirm after these three contributions:

- Advancements in neural topic model helped to improve node clustering results.
- Graph neural network can be efficiently coupled with a neural topic model to provide an intelligible visualisation as well as an efficient node clustering methodology.
- Using a graph convolutional network to encode the node connections into node positions enables to simultaneously estimate a block model as well as a positional model.

Interesting future works:

- (short-term) Deriving a model selection criterion for Deep-LPBM.
- (short-term) Performing an extensive benchmark and a real data analysis to evaluate Deep-LPBM efficiency.
- (medium-term) Taking into account the number of exchanges/interactions.
- (medium-term) Using mini-batches for networks with textual edges.

Contributions

Our first contribution was shared with the community through a publication in an international peer-reviewed journal:

- **Embedded topics in the stochastic block model**, joint work with Pierre Latouche and Charles Bouveyron, *Statistics and Computing (2023)*.

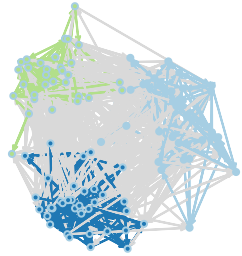
The second contribution is under revision at the Scandinavian Journal of Statistics:

- **The Deep Latent Position Topic Model for Clustering and Representation of Networks with Textual Edges**, joint work with Pierre Latouche and Charles Bouveyron, *Pre-print hal-04068665 (2023)*.

The last work will be submitted after conducting the final numerical experiments:

- **The Deep Latent Position Block Model**, joint work with Pierre Latouche and Charles Bouveyron.

Thank you all for your attention



References

- Airoldi, Edoardo M. et al. (2008). *Mixed Membership Stochastic Blockmodels*. In: Journal of Machine Learning Research, Vol. 9, No. 65, pp. 1981–2014. URL: http://jmlr.org/papers/v9/airoldi08a.html.
- Biernacki, Christophe, Gilles Celeux, and Gérard Govaert (2000). *Assessing a mixture model for clustering with the integrated completed likelihood*. In: IEEE transactions on pattern analysis and machine intelligence, Vol. 22, No. 7, pp. 719–725.
- Bouveyron, Charles, Pierre Latouche, and Rawya Zreik (2018). *The stochastic topic block model for the clustering of vertices in networks with textual edges*. In: Statistics and Computing, Vol. 28, No. 1, pp. 11–31.
- Côme, Etienne and Pierre Latouche (2015). *Model selection and clustering in stochastic block models based on the exact integrated complete data likelihood*. In: Statistical Modelling, Vol. 15, No. 6, pp. 564–589. DOI: 10.1177/1471082X15577017.
- Daudin, J-J, Franck Picard, and Stéphane Robin (2008). *A mixture model for random graphs*. In: Statistics and computing, Vol. 18, No. 2, pp. 173–183.

- Dieng, Adji B, Francisco JR Ruiz, and David M Blei (2020). *Topic modeling in embedding spaces*. In: Transactions of the Association for Computational Linguistics, Vol. 8, pp. 439–453.
- Gopalan, Prem K and David M Blei (2013). *Efficient discovery of overlapping communities in massive networks*. In: Proceedings of the National Academy of Sciences, Vol. 110, No. 36, pp. 14534–14539.
- Handcock, Mark S, Adrian E Raftery, and Jeremy M Tantrum (2007). *Model-based clustering for social networks*. In: Journal of the Royal Statistical Society: Series A (Statistics in Society), Vol. 170, No. 2, pp. 301–354.
- Hoff, Peter D, Adrian E Raftery, and Mark S Handcock (2002). *Latent space approaches to social network analysis*. In: Journal of the american Statistical association, Vol. 97, No. 460, pp. 1090–1098.
- Kingma, Diederik P and Max Welling (2014). *Auto-Encoding Variational Bayes*. arXiv: 1312.6114 [stat.ML].
- Kipf, Thomas N and Max Welling (2016). *Variational graph auto-encoders*. arXiv: 1611.07308 [stat.ML].
- Kipf, Thomas N. and Max Welling (2017). *Semi-Supervised Classification with Graph Convolutional Networks*. In: International Conference on Learning Representations. URL: https://openreview.net/forum?id=SJU4ayYgl.

- Latouche, Pierre, Etienne Birmele, and Christophe Ambroise (2012). *Variational Bayesian inference and complexity control for stochastic block models*. In: Statistical Modelling, Vol. 12, No. 1, pp. 93–115.
- Liang, Dingge et al. (2022). *Deep latent position model for node clustering in graphs*. In: The 30th European Symposium on Artificial Neural Networks (ESANN 2022).
- McDaid, Aaron F et al. (2013). *Improved Bayesian inference for the stochastic block model with application to large networks*. In: Computational Statistics & Data Analysis, Vol. 60, pp. 12–31.
- Rezende, Danilo Jimenez, Shakir Mohamed, and Daan Wierstra (2014). *Stochastic backpropagation and approximate inference in deep generative models*. In: International conference on machine learning. Proceedings of Machine Learning Research, pp. 1278–1286.

Reparametrisation trick for ETSBM

$$\mathbb{E}_R\left[\log p(\mathbf{W} \mid \mathbf{C}, \mathbf{A}, \mathbf{Y}, \boldsymbol{\alpha}, \boldsymbol{\rho})\right] = \sum_{i \neq j}^{M} \sum_{q,r}^{Q} A_{ij} \tau_{iq} \tau_{jr} \mathbb{E}_{R_\nu}\Big[\underbrace{\log p(w_{ij} \mid Y_{qr}, \boldsymbol{\alpha}, \boldsymbol{\rho})}_{T_{ij}^{Y_{qr}}}\Big],$$

with,

$$T_{ij}^{Y_{qr}} = \sum_{d=1}^{D_{ij}} \sum_{n=1}^{N_{id}^d} \sum_{v=1}^{V} w_{ij}^{dnv} \log\left(\sum_{k=1}^{K} \theta_{qrk} \beta_{kv}\right). \tag{15}$$

Using $Y_{qr} = \mu_{qr}(\tau, \nu) + \sigma_{qr}(\tau, \nu)\epsilon$, $\epsilon \sim \mathcal{N}(0_K, \mathrm{I}_K)$, we can approximate $T_{ij}^{Y_{qr}}$ using a Monte-Carlo estimator such that:

$$\epsilon^s \sim \mathcal{N}(0, \mathbf{I}_K), \quad Y_{qr}^s = \mu_{qr}(\tau, \nu) + \sigma_{qr}(\tau, \nu) \odot \epsilon^s, \quad \theta_{qr}^s = \mathrm{softmax}(Y_{qr}^s).$$

with $\odot$ denoting the Hadamard product. Thus, for each pair of nodes (i, j) and pair of clusters (q, r), the estimate is given by:

$$\hat{T}_{ij}^{qr} = S^{-1} \sum_{s=1}^{S} T_{ij}^{Y_{qr}^s}.$$

Noise in the synthetic datasets

- node i (j resp.) in cluster q (r resp.)
- topic proportion $\theta^{\star}_{qr} = (0, \ldots, 01, 0 \ldots, 0)$ with 1 on the corresponding topic
- $\zeta = 0$: pure topic, $\zeta = 1$: uniform distribution over topics
- $\eta = 0.1$ instead of 0.25

$$\theta_{qr} = (1 - \zeta)\theta^{\star}_{qr} + \zeta * \left(\frac{1}{K}, \ldots, \frac{1}{K}\right)^{\top}. \tag{16}$$

Effect of the initialisation on ETSBM

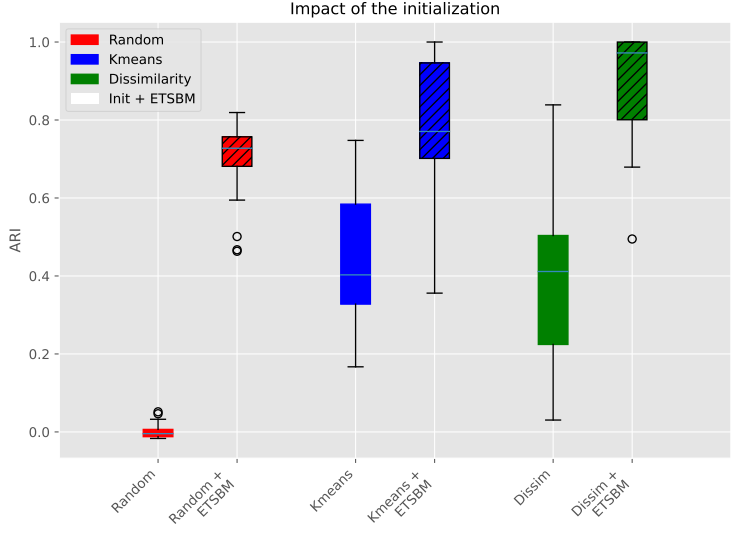


Figure 18: ARI of the initialisation (without stripe) and of ETSBM clustering obtained on 50 Scenario C networks following the *Hard* setting.

Analysis of the Tweets published or "retweeted" prior to the 2022 French presidential election

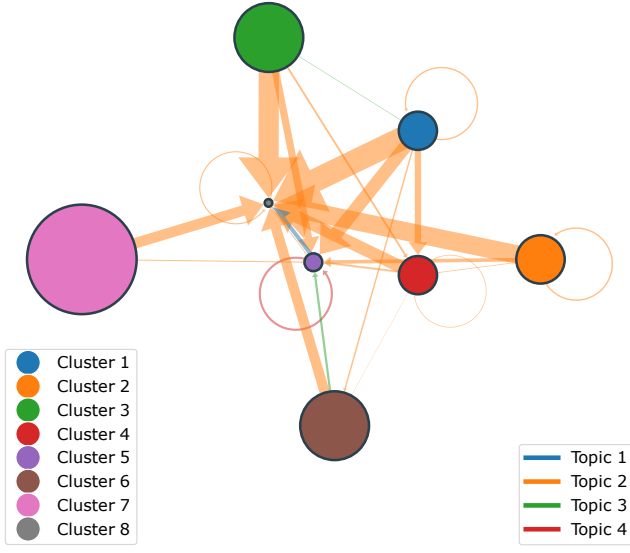


Figure 19: SBM clusters of the twitter accounts and ETM topics of the tweets .

Figure 20: The most important words of the topics presented in the meta-graph above for ETM and SBM estimated separately.

IC2L model selection results for different triplets (K, P, Q)

	$K = 2$	$\mathbf{K = 3}$	$K = 4$	$K = 5$	$K = 6$
$Q = 2$	0	0	0	0	0
$Q = 3$	0	0	0	0	0
$\mathbf{Q = 4}$	0	10	0	0	0
$Q = 5$	0	0	0	0	0
$Q = 6$	0	0	0	0	0

Table 6: Number of times a triplet (K, P, Q) is associated with the highest IC2L over 10 graphs simulated according to Scenario C ($Q^\star = 4$ and $K^\star = 3$). All the models with the highest IC2L value correspond to $P = 2$. Therefore, only the table corresponding to this value is shown.

Model selection criterion for Deep-LPTM

$$
\begin{aligned}
\widehat{ICL}(\mathcal{M}, Q, K, P) = & \max_{\kappa} \log p(\mathbf{A} \mid \hat{\mathbf{Z}}, \kappa, \mathcal{M}) - \frac{1}{2}\log(N(N-1)) \\
& + \max_{\mu,\sigma} \log p(\hat{\mathbf{Z}} \mid \hat{\mathbf{C}}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \mathcal{M}, Q, P) - \frac{QP+Q}{2}\log(N) \\
& + \max_{\boldsymbol{\rho},\boldsymbol{\alpha}} \log p(\mathbf{W} \mid \mathbf{A}, \hat{\mathbf{Y}}, \boldsymbol{\rho}, \boldsymbol{\alpha}, \mathcal{M}) - \frac{VL+KL}{2}\log(M) \\
& + \max_{\mathbf{m},\mathbf{s}} \log p(\hat{\mathbf{Y}} \mid \mathbf{A}, \hat{\mathbf{C}}, \mathbf{m}, \mathbf{s}, \mathcal{M}, K) - \frac{Q^2K+Q^2}{2}\log(M) \\
& + \max_{\gamma} \log p(\hat{\mathbf{C}} \mid \gamma, \mathcal{M}, Q) - \frac{Q-1}{2}\log(N),
\end{aligned}
$$

with $\hat{\mathbf{Z}}$, $\hat{\mathbf{Y}}$ and, $\hat{\mathbf{C}}$ the maximum-a-posteriori estimates, and

$$
\begin{aligned}
\Omega(\mathcal{M}, Q, K, P) = & \frac{1}{2}\log(N(N-1)) \\
& + \frac{Q(P+2)-1}{2}\log(N) + \frac{L(V+K)+Q^2(K+1)}{2}\log(M).
\end{aligned}
$$

Impact of the initialisation in Deep-LPTM

		Scenario A	Scenario B	Scenario C
Hard	Random init	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
	Dissimilarity init	0.31 ± 0.14	1.00 ± 0.00	0.38 ± 0.24
	Deep-LPTM random	0.80 ± 0.21	0.95 ± 0.05	0.47 ± 0.02
	Deep-LPTM dissim	0.99 ± 0.02	$\mathbf{1.00 \pm 0.00}$	$\mathbf{0.89 \pm 0.15}$
	Deep-LPTM - PT random	0.95 ± 0.05	0.73 ± 0.30	0.45 ± 0.04
	Deep-LPTM - PT dissim	$\mathbf{1.00 \pm 0.01}$	$\mathbf{1.00 \pm 0.00}$	0.85 ± 0.18

Table 7: Adjusted rand index (ARI) of the initialisations and the results of Deep-LPTM in terms of node clustering, without and with pre-trained embeddings (denoted PT in that case). ARI is averaged over 10 graphs, for each scenario and difficulty.

Mini-batch in networks with textual edges

Based on notations from Gopalan and Blei (2013), we denote

- $g(i, j)$ the edge density

We have to incorporate the sampling strategy into the topic modelling inference. Sampling a pair of nodes (i, j) according to g, we have:

$$\mathbb{E}_R \left[\log p(\mathbf{A} \mid \mathbf{Z}, \kappa)\right] = \mathbb{E}_g \left[\frac{1}{g(i,j)} \mathbb{E}_R \left[\log p(A_{i,j} \mid Z_i, Z_j, \kappa)\right]\right],$$

$$\mathbb{E}_R \left[\log p(\mathbf{W} \mid \mathbf{A}, \mathbf{Y}, \boldsymbol{\rho}, \boldsymbol{\alpha})\right] = \mathbb{E}_g \left[\frac{1}{g(i,j)} \mathbb{E}_R \left[\log p(W_{i,j} \mid A_{i,j} Y_{i,j}, \boldsymbol{\rho}, \boldsymbol{\alpha})\right]\right].$$

Mini-batch in networks with textual edges

An unbiased expectation of the ELBO:

$$
\begin{aligned}
\mathscr{L}(R(\cdot);\Theta) = & \sum_{i=1}^{N} \mathbb{E}_R\left[\log p(C_i \mid \gamma)\right] - \mathbb{E}_R\left[\log R(C_i)\right] \\
& + \sum_{i=1}^{N} \mathbb{E}_R\left[\log p(Z_i \mid C_i, \boldsymbol{\mu}, \boldsymbol{\sigma})\right] - \mathbb{E}_R\left[\log R(Z_i \mid \mathbf{A})\right] \\
& + \mathbb{E}_g\left[\frac{1}{g(I,J)}\Big(\mathbb{E}_R\left[\log p(A_{I,J} \mid Z_i, Z_j, \kappa)\right] + \mathbb{E}_R\left[\log p(W_{I,J} \mid A_{I,J}, Y_{I,J}, \boldsymbol{\rho}, \boldsymbol{\alpha})\right]\Big)\right] \\
& + \mathbb{E}_g\left[\frac{1}{g(I,J)}\Big(\mathbb{E}_R\left[\log p(Y_{I,J} \mid \mathbf{A}, C_I, C_J, \mathbf{m}, \mathbf{s})\right] - \mathbb{E}_R\left[\log R(Y_{I,J} \mid \mathbf{A}, W_{I,J})\right]\Big)\right].
\end{aligned}
\tag{17}
$$

Therefore, we can compute an unbiased estimator of the gradient with respect to m_{qr}. To avoid cumbersome notations, the last term in Equation (17) will be denoted:

$$
\mathrm{KL}_{IJ}^{qr} = \mathrm{KL}\Big(\mathcal{N}_K\left(\mu_{\phi_Y}(W_{I,J} \mid \mathbf{A}), \sigma_{\phi_Y}(W_{I,J} \mid \mathbf{A})\right) \,||\, \mathcal{N}_K\left(m_{qr}, s_{qr}^2 \mathbf{I}_K\right)\Big), \tag{18}
$$

Mini-batch in networks with textual edges

To illustrate the new optimisation strategy, we detail the computations to update the parameter m_{qr}. Noting that the only term depending on m_{qr} in Equation (17) is the one detailed in Equation (18), the partial derivative of the ELBO with respect to m_{qr} is given by:

$$\frac{\partial}{\partial m_{qr}} \mathscr{L}(R(\cdot); \Theta) = -\mathbb{E}_g \left[\frac{\tau_{Iq}\tau_{Jr} A_{I,J}}{g(I,J)} \frac{\partial}{\partial m_{qr}} \mathrm{KL}_{IJ}^{qr} \right] = -\mathbb{E}_g \left[\frac{\tau_{Iq}\tau_{Jr} A_{I,J}}{g(I,J)} \frac{1}{\sigma_{qr}^2} (m_{qr} - \mu_{\phi_Y}(W_{I,J})) \right].$$