

Sommaire et ressources pour le mémoire : Modélisation de Pandémies par GNNs

30 mai 2025

1 Éléments Essentiels du Mémoire

1.1 Introduction

- **Contexte :**
 - Enjeux des pandémies (dynamique spatio-temporelle, interdépendances régionales)
 - Limites des modèles classiques (SIR/SEIR)
- **Problématique :**
 - Exploitation des réseaux complexes et de l'hétérogénéité spatiale
- **Objectifs :**
 - Framework GNN avec *transfer learning*
 - Validation sur données réelles

1.2 Revue de Littérature

- Épidémiologie Mathématique (modèles compartimentaux)
- Deep Learning pour séries temporelles (LSTM, Transformers)
- Graph Neural Networks (GCN, GAT, MPNN)
- Transfer Learning (adaptation de domaine)

1.3 Modèle Proposé : *PandemicGNN-Transfer*

- **Architecture :**
 - Module spatial : GAT pour agrégation des voisins
 - Module temporel : LSTM couplée
 - Mécanisme de transfert : adaptation de domaine
- **Formalisation Mathématique :**

$$h_v^{(t)} = \text{LSTM} \left(\left[x_v^{(t)}, \sum_{u \in \mathcal{N}(v)} \alpha_{vu} h_u^{(t-1)} \right] \right)$$
$$\alpha_{vu} = \text{attention}(Wh_u, Wh_v)$$
$$\mathcal{L} = \mathcal{L}_{\text{MSE}} + \lambda \mathcal{L}_{\text{transfert}}$$

1.4 Expérimentations & Résultats

- **Données :**
 - Épidémiologiques (OMS, CDC (Centers for Disease Control and Prevention), Johns Hopkins)
 - Graphes de mobilité (Google, OSM (OpenStreetMap))
 - Contextuelles (démographie, politiques)
- **Résultats (espérés) :**
 - +15% de précision vs LSTM
 - Identification des régions super-propagatrices

1.5 Discussions

1.5.1 Robustesse

1.5.2 Limites & Perspectives

1.6 Conclusion

Références

2 Ressources Informatiques

2.1 Matériel

- **GPU** : NVIDIA $\geq 8\text{Go}$ VRAM (RTX 3080+)
- **CPU/RAM** : 8 cœurs + 32Go RAM
- **Stockage** : SSD $\geq 512\text{Go}$
- **Cloud** : AWS/GCP/Azure (V100/A100)

2.2 Logiciels & Bibliothèques

TABLE 1 – Bibliothèques clés

Catégorie	Bibliothèques
GNNs	PyTorch Geometric, DGL
Deep Learning	PyTorch, TensorFlow
Traitement de données	Pandas, NumPy
Visualisation	Matplotlib, Plotly

2.3 Données & Stockage

- Sources : Google Mobility, OMS, OpenStreetMap
- Espace requis : $\geq 50\text{Go}$
- Formats : Parquet/HDF5 pour compression

2.4 Gestion de Projet

- Versioning : Git + DVC
 - Conteneurisation : Docker
- ```
FROM pytorch/pytorch:2.0.1-cuda11.7
RUN pip install torch-geometric pandas
```

## 2.5 Optimisation

TABLE 2 – Estimation des coûts

| Phase          | Temps (GPU V100) | Coût (AWS) |
|----------------|------------------|------------|
| Pré-traitement | 2-6h             | ~5\$       |
| Entraînement   | 10-30h           | 30-90\$    |
| Inférence      | <1h              | <2\$       |

## 2.6 Solutions Économiques

- Google Colab Pro (~10\$/mois)
- Techniques d'optimisation :
  - Mini-batching
  - Mixed Precision (FP16)