

Graphes dynamiques pour les Graph Neural Networks et la recommandation

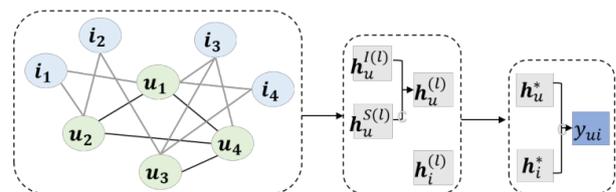
Encadrants : Raphaël Fournier-S'niehotta et Nicolas Thome (équipe Vertigo)

Introduction

L'apprentissage profond (*Deep Learning*) a montré durant la dernière décennie une efficacité impressionnante pour résoudre de nombreux problèmes (vision par ordinateur, reconnaissance de la parole, imagerie médicale, systèmes de recommandation, etc.). Les récents modèles attentionnels ont aussi chamboulé le traitement des langues.

Les graphes sont des structures utilisées pour la modélisation de problèmes dans de nombreux domaines (des transports à la biologie des protéines). L'analyse des réseaux connaît depuis une vingtaine d'années un essor nouveau, qui a produit des résultats importants, tant en analyse de cas précis qu'en termes de modélisation de grands graphes. Cependant, les graphes apportent une complexité que n'ont pas les données classiques de l'apprentissage. En effet, si l'ordre des pixels dans une image est fixe et immuable, il n'existe par exemple pas d'ordre naturel pour parcourir les voisins d'un nœud dans un graphe. Des architectures reposant sur la convolution (et donc l'équivariance) ne sont pas utilisables directement et doivent être adaptées.

Diverses techniques ont été récemment mises au point pour surmonter ces obstacles, à commencer par l'apprentissage de représentations pour chaque nœud (*embeddings* [GL16]). Ils ont évolué en Graph Convolutional Networks [KW16] et Graph Neural Networks [BZSL13, SGT⁺08], avec des modèles centrés autour du passage de messages entre nœuds, pour agréger des représentations du voisinage d'un nœud de plus en plus élargi couche après couche. Des travaux indépendants dans le domaine du signal, en Graph Signal Processing, ont également convergé vers ces modèles.



Contexte national et international

Dans de nombreuses situations, les graphes représentent des données qui sont, intrinsèquement, dynamiques (communications entre personnes, échanges financiers, etc.). Pourtant, faute de consensus autour de la modélisation, ce sont souvent des graphes statiques qui sont employés, agrégeant des périodes de temps plus ou moins importantes et détruisant ainsi de l'information. Les Graph Neural Networks ont, eux aussi, été conçus à partir de graphes statiques (éventuellement des séries de *snapshots*). Et ce n'est que tout récemment que des modèles ont émergé pour prendre en compte un temps *continu* et ont proposé des applications [TFBZ19, MMMC20].

Plusieurs questions sont donc ouvertes, pour avancer vers de meilleurs modèles de GNN capables d'intégrer la dimension temporelle des données, de combiner efficacement la diffusion spatiale et la diffusion temporelle de l'information. Le passage à grande échelle, l'explicabilité des modèles, ainsi qu'une standardisation de l'évaluation sont aussi des enjeux des années à venir dans le domaine.

Objectif

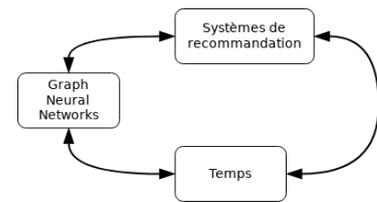
L'objectif de la thèse est de proposer des modèles de *graph neural networks* permettant de prendre en compte explicitement une dimension temporelle des données.

Si l'ambition formelle est générale, nous envisageons avant tout d'évaluer les modèles dans un contexte de systèmes de recommandation. Ces systèmes visent à filtrer des ensembles d'éléments (d'un ensemble de grande taille) pour les proposer à des utilisateurs (souvent nombreux), en fonction de préférences (inférées

par les actions passées des utilisateurs sur les éléments du système) : les données peuvent être vues comme un grand graphe biparti *utilisateurs-items*. La dimension temporelle est inhérente aux données, un *utilisateur* consomme un *item* à un *instant t*.

La recommandation consiste à faire de la prédiction de liens dans le graphe. Les modèles dominants du domaine reposent grandement sur de la factorisation de matrices (filtrage collaboratif centré *utilisateurs* ou *items*), la dimension temporelle des données est souvent laissée de côté dans les algorithmes de recommandation, au profit d'une modélisation simplifiée.

Dans ce contexte, l'agrégation de données amène à considérer comme identique d'avoir consommé un contenu au début ou à la fin de la capture des données, et les arrivées et départs d'utilisateurs et produits sont invisibilisées. Il existe cependant des modèles performants et assez complets [Kor09].



Approche

Axe 1 : approche expérimentale de l'état de l'art

Le premier axe de la thèse consistera à évaluer des méthodes de référence pour la recommandation reposant sur des GNNs (GraphRec, NGCF, PinSage), et en particulier les approches incorporant une dimension temporelle comme [RCF⁺20]. L'approche modulaire de ce framework permet de décomposer les problématiques liées aux GNNs dans un contexte dynamique : représentation de la mémoire d'un nœud, gestion de la "fraîcheur" de cette mémoire, choix des fonctions de passage de messages et d'agrégation, mise à jour des *embeddings*. En particulier, l'encodage du temps pour ces *embeddings* pourra être étudié minutieusement, pour faire jauger des évolutions possibles de Time2Vec [XRK⁺20, KGE⁺19].

Axe 2 : graphes dynamiques et GNN

Le deuxième axe de la thèse consistera d'abord en une analyse des modèles de graphes dynamiques existants. Nous aborderons d'abord des "séquences d'instantanés" (*snapshots*), mais envisagerons rapidement des approches reposant sur un temps continu. En effet, dans de nombreux contextes, perdre de l'information avec un modèle séquentiel n'est pas satisfaisant, et l'on doit viser des approches reposant sur un temps continu. Des travaux récents [LVM18], réutilisés par l'équipe [VFS19], ont apporté un nouveau modèle (*stream graph*) que l'on peut encapsuler dans une architecture GNN.

Parallèlement à ce travail, des améliorations des modèles existants seront envisagées : en effet, les quelques modèles GNN à temps continu ne considèrent que des *ajouts de liens*, et ne sont pas encore en mesure de prendre en compte des *suppressions de liens*, et des *ajouts et suppressions de nœuds*, pourtant essentiels dans l'évolution de telles structures.

Axe 3 : évaluation de GNN temporels pour la recommandation

Dans un troisième temps, la thèse abordera la question de l'évaluation des modèles conçus, dans un contexte applicatif centré sur la recommandation. La prise en compte de la dynamique en recommandation est importante, et les apports de modèles GNN pour des systèmes à grande échelle pourraient être significatifs (gestion efficace de l'évolution des préférences d'un utilisateur, sur de longues périodes, par exemple).

Mettre en place un dispositif expérimental rigoureux dans un tel contexte sera aussi une contribution pour le domaine des GNNs en général, puisque la plupart des modèles sont étudiés sur des jeux de données différents, rendant les comparaisons délicates.

Échéancier

Les premiers mois de la thèse seront consacrés à l'étude bibliographique des différents aspects du sujet : GNN pour la recommandation, GNN et dynamique, évaluation de la dimension temporelle dans la recommandation.

L'axe 1, qui correspond à une prise en main expérimentale des modèles de l'état de l'art, sera abordé de façon complémentaire (débutant après quelques mois et se poursuivant en parallèle de l'étude bibliographique).

L'axe 2 commencera en fin de première année et devrait durer une dizaine de mois. L'axe 3, sur l'évaluation applicative dans un contexte de recommandation, occupera les 10 à 12 mois suivants. Les derniers mois de la thèse seront consacrés à la rédaction du manuscrit et la soutenance.

Environnement scientifique

L'équipe Vertigo du CNAM Paris (<http://cedric.cnam.fr/lab/equipes/vertigo/>) étudie principalement la fouille de données (*data mining*) appliquée à des données complexes, avec des spécialistes de l'apprentissage profond [DMTC17] mais aussi de la fouille de graphes pour la recommandation, notamment dans un contexte temporel [VFS19].

Profil

La candidate ou le candidat doit avoir de solides bases en informatique et mathématiques appliquées (M2 ou ingénieur), en particulier l'apprentissage automatique et idéalement de bonnes notions de graphes. La maîtrise à un bon niveau d'un langage de programmation (comme Python) est indispensable.

Les candidatures s'effectuent par mail aux adresses fournier@cnam.fr et nicolas.thome@cnam.fr, avec au moins un CV et une description du profil (ou lettre de motivation).

Références

- [BZSL13] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv :1312.6203*, 2013.
- [DMTC17] Thibaut Durand, Taylor Mordan, Nicolas Thome, and Matthieu Cord. Wildcat : Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 642–651, 2017.
- [GL16] Aditya Grover and Jure Leskovec. node2vec : Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.
- [KGE⁺19] Seyed Mehran Kazemi, Rishab Goel, Sepehr Eghbali, Janahan Ramanan, Jaspreet Sahota, Sanjay Thakur, Stella Wu, Cathal Smyth, Pascal Poupard, and Marcus Brubaker. Time2vec : Learning a vector representation of time. *arXiv preprint arXiv :1907.05321*, 2019.
- [Kor09] Yehuda Koren. Collaborative filtering with temporal dynamics. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 447–456, 2009.
- [KW16] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv :1609.02907*, 2016.
- [LVM18] Matthieu Latapy, Tiphaine Viard, and Clémence Magnien. Stream graphs and link streams for the modeling of interactions over time. *Social Network Analysis and Mining*, 8(1) :1–29, 2018.
- [MMMC20] Eli A Meir, Haggai Maron, Shie Mannor, and Gal Chechik. How to stop epidemics : controlling graph dynamics with reinforcement learning and graph neural networks. *arXiv preprint arXiv :2010.05313*, 2020.
- [RCF⁺20] Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael Bronstein. Temporal graph networks for deep learning on dynamic graphs. *arXiv preprint arXiv :2006.10637*, 2020.
- [SGT⁺08] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1) :61–80, 2008.
- [TFBZ19] Rakshit Trivedi, Mehrdad Farajtabar, Prasenjeet Biswal, and Hongyuan Zha. Dyrep : Learning representations over dynamic graphs. In *International Conference on Learning Representations*, 2019.
- [VFS19] Tiphaine Viard and Raphaël Fournier-S'niehotta. Augmenting content-based rating prediction with link stream features. *Computer Networks*, 150 :127–133, 2019.
- [XRK⁺20] Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. Inductive representation learning on temporal graphs. In *ICLR*, 2020.