

Friend Recommendation in social network

Nous sommes à une époque où les sciences de la connaissance (en particulier l'apprentissage automatique et l'intelligence artificielle) sont à leur apogée et intégrées dans tous les aspects ou domaines qui nous entourent. Les sites web les plus populaires utilisés par toutes les tranches d'âge, à savoir les sites de réseaux sociaux et les sites de e-commerce, utilisent une toute autre idée du machine learning, notamment pour faire des suggestions ou plutôt des recommandations à leurs utilisateurs afin de les attirer selon certains critères. Ce type de système est souvent appelé système de recommandation. Tous les sites populaires comme Facebook, Twitter, Snapchat, Instagram, Amazon, etc. Utilisez ce système de parrainage pour attirer leurs utilisateurs et les faire payer plus longtemps pour leur application et leur site Web. Les systèmes de recommandation sont généralement divisés en grandes catégories dans les catégories suivantes :

- Système de recommandation d'amis,
- Système de recommandation de photos
- Système de recommandation de produits, etc.

Ici, nous avons tendance à discuter principalement du système de recommandation d'amis. Cependant, avant de poursuivre, nous devons toujours appréhender un peu plus le système de recommandation et le système contextuel.

Les systèmes de recommandation d'amis utilisent des informations sur les relations sociales entre les utilisateurs pour recommander des amis potentiels. Il peut utiliser des informations telles que des amis communs, des groupes auxquels appartient un utilisateur ou des pages qu'il aime. D'autre part, les systèmes de recommandation contextuels utilisent des informations sur les utilisateurs et leurs contextes pour recommander des articles ou des produits pertinents. Il peut utiliser des informations telles que l'emplacement de l'utilisateur, la météo, l'heure, les préférences d'achat ou les recherches récentes. Ces deux systèmes peuvent être combinés pour recommander des amis en fonction du contexte de l'utilisateur.

Dans notre cas, nous avons traité le système de recommandation de Facebook. En fait ce type de système de recommandation d'amis est traditionnellement réalisé à l'aide de méthodes telles que l'heuristique basée sur des graphes. Les progrès récents des réseaux de neurones graphiques (GNN) offrent de nouvelles approches pour cette tâche, car cette famille de modèles peut capturer plus efficacement les similitudes entre les utilisateurs. Dans ce notebook, nous expliquerons comment fonctionne le modèle basé sur GNN et créerons étape par étape un modèle de prédiction d'amis.

C'est quoi le GNN ?

Comment peut-on l'implémenter pour créer un système de recommandation d'amis ?

Un système de recommandation d'amis peut être créé à l'aide d'un réseau neuronal graphique (GNN). Les GNN permettent de traiter des données structurées sous forme de graphes, comme les relations entre utilisateurs de plateformes sociales. Les GNN peuvent apprendre à reconnaître des modèles dans les relations avec les utilisateurs et utiliser ces informations pour recommander des amis potentiels. Il existe différentes architectures GNN disponibles pour de telles tâches, telles que les réseaux de neurones convolutifs de graphes (GCN) ou les réseaux de neurones récurrents de graphes (GRN). Avoir une grande quantité de données est très important pour former efficacement un modèle GNN pour la recommandation d'amis.

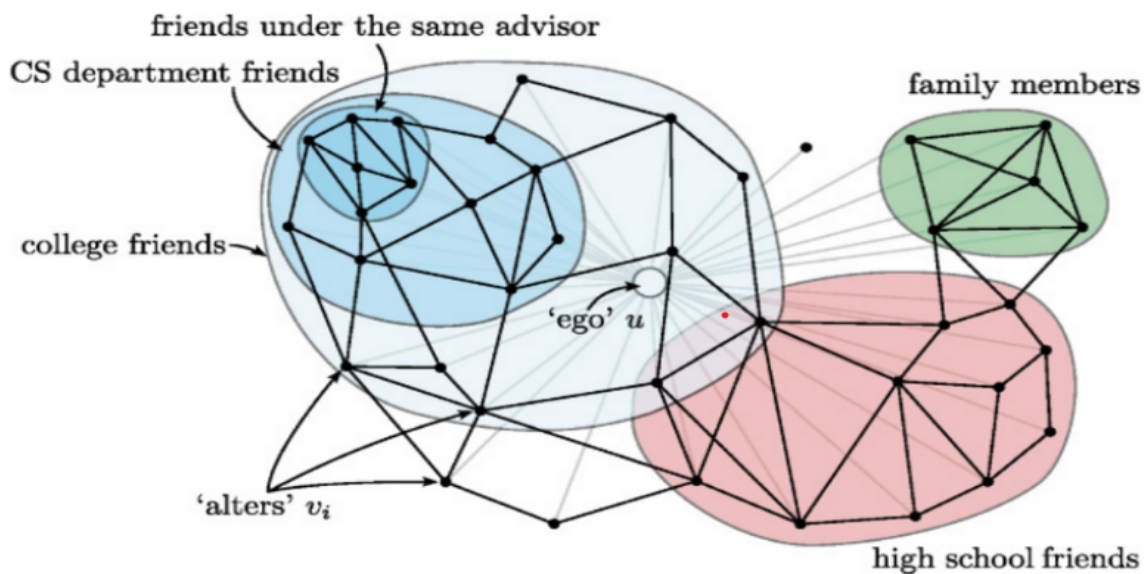
Un système de recommandation de réseau social utilisant des GNN nécessite généralement un ensemble de données comprenant des informations sur les utilisateurs et leurs connexions dans le réseau. Le jeu de données peut être collecté à partir de diverses sources, telles que les API de réseaux sociaux accessibles au public ou en extrayant des données du site Web du réseau social. La composition de l'ensemble de données dépendra de la tâche spécifique et des informations disponibles, mais il comprend généralement les types de données suivants :

- Informations sur l'utilisateur : Il peut s'agir d'informations démographiques, telles que l'âge, le sexe, la localisation et les centres d'intérêt, ainsi que d'informations sur l'activité, telles que le nombre de publications, de commentaires et d'appréciations.
- Structure du réseau : Il s'agit d'informations sur les liens entre les utilisateurs, par exemple qui suit qui, qui est ami avec qui et qui fait partie d'un groupe.
- Interactions : Il s'agit d'informations sur la façon dont les utilisateurs interagissent les uns avec les autres, comme le nombre de messages échangés, de commentaires sur les messages et d'appréciations sur les messages.

Description du DataSet

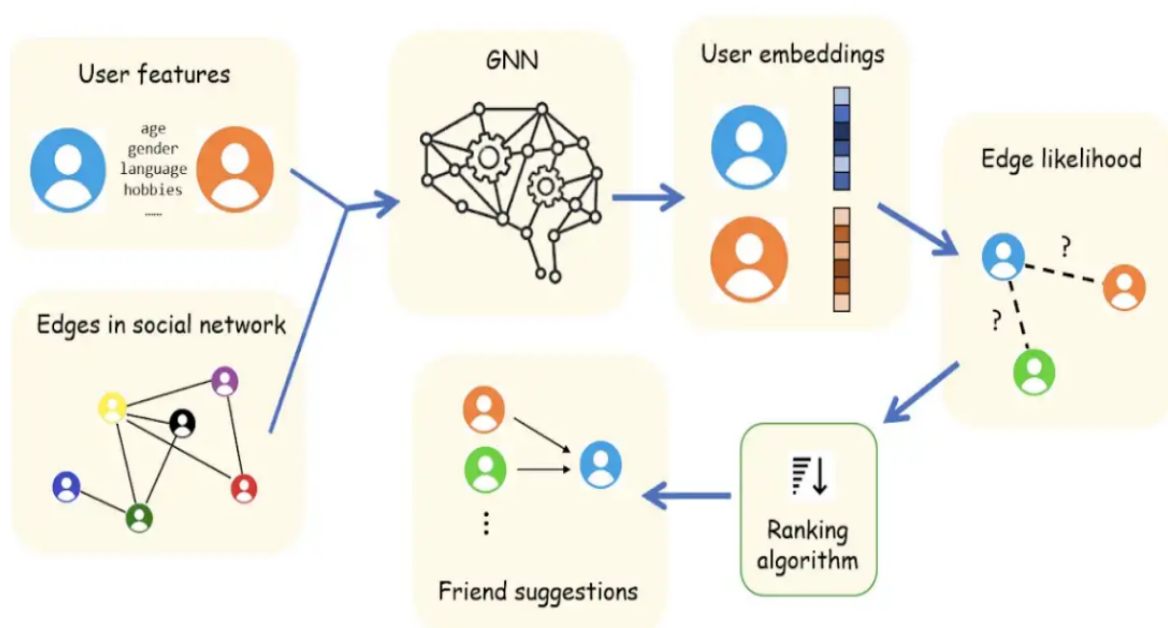
Nous utilisons les données de Facebook dans notre projet. Le jeu de données comprend des caractéristiques de nœuds (profils), des cercles et des réseaux d'ego.

“Ego” fait référence au propriétaire du réseau. L'utilisateur ego a la possibilité de créer des cercles basés sur les connexions et les caractéristiques partagées par les autres utilisateurs. Dans la figure ci-dessus, l'utilisateur central (ego) est ami avec tous les autres utilisateurs du réseau. Cinq cercles sont notés sur le graphique : les amis du même conseiller, les amis du département CS, les amis du collège, les membres de la famille et les amis du lycée. Il est possible que les cercles glissent. Pour des raisons de simplicité, nous n'utilisons qu'un seul des réseaux d'ego pour l'ensemble des données. Il y a 5038 arêtes et 347 nœuds dans le graphe. Chaque nœud est représenté par un vecteur caractéristique à 224 dimensions. Le code qui suit transmet les données et facilite leur prétraitement.



Avant d'utiliser l'ensemble de données pour entraîner un GNN, il peut être nécessaire de le prétraiter pour s'assurer qu'il est dans le format approprié pour le modèle. Les données de réseau social sont converties en un graphe, où les nœuds représentent les utilisateurs et les arêtes représentent les relations d'amitié entre eux. Les caractéristiques des nœuds, telles que les intérêts, les centres d'intérêt et les informations de profil, sont également extraites et utilisées comme entrées pour le réseau de neurones.

Les étapes de la prédiction des liens dans le GNN :



Prédiction des liens

La prédiction de liens consiste à prédire les liens manquants ou futurs dans un graphe sur la base de la structure existante et des caractéristiques des nœuds. L'objectif est d'identifier les relations susceptibles d'exister entre les nœuds qui n'ont actuellement pas de lien. Dans ce cas, nous pouvons modéliser cette tâche de prédiction de liens en estimant un score de "vraisemblance" à chaque arête potentielle. Une manière simple d'obtenir ce score est de prendre le produit scalaire entre les incorporations apprises de deux nœuds.

Rank potential friends

Pour un utilisateur donné, trier les autres utilisateurs du graphe par le score de lien prédit et sélectionner les utilisateurs les mieux classés comme amis potentiels.

Train-Test split

Comme nous n'avons pas accès à des informations sur le temps pour l'ensemble de données Facebook. Nous avons divisé 30% des liens pour les tests, et 70% pour l'entraînement de manière aléatoire.

Métrique basée sur la prédiction

Nous avons utilisé dans ce projet l'Area Under ROC Curve (AUC) comme exemple de mesure basée sur la classification. Sur la courbe ROC (Receiver Operating Characteristics), le taux de faux positifs (FPR) est représenté sur l'axe des x, tandis que le taux de vrais positifs (TPR) est représenté sur l'axe des y. Lorsque le seuil de classification est de 0, le FPR et le TPR sont tous deux égaux à 0, puisque tout est étiqueté "négatif". Lorsque nous augmentons le seuil jusqu'à 1, le modèle classe de plus en plus de points de données comme "positifs", ce qui entraîne une augmentation du TPR et du FPR.

Performance de notre modèle

Nous avons parcouru tous les composants de notre pipeline d'apprentissage automatique pour la recommandation d'amis, des caractéristiques d'entrée au modèle GNN, en passant par l'algorithme du classement des amis, les mesures d'évaluation, et nous avons obtenu avec GCN un score AUC de 0.80.

Bibliographie

- [1] McAuley, J. J., & Leskovec, J. (2012). Learning to discover social circles in ego networks. In NIPS (Vol. 2012, pp. 548–56).
- [2] Mcauley, J., & Leskovec, J. (2014). Discovering social circles in ego networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 8(1), 1–28.
- [3] Hamilton, W., Ying, Z., & Leskovec, J. (2017). Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.
- [4] Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
- [5] Wang, M., Zheng, D., Ye, Z., Gan, Q., Li, M., Song, X., ... & Zhang, Z. (2019). Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv preprint arXiv:1909.01315*.