

BIML GNN : Prédiction de lien

VGAE et autres joyeusetés

Bonhoure Timothé, Martinez Christophe

30 octobre 2023

Table des matières

1	Méthode	2
2	Dropout	2
2.1	Méthode	2
2.2	Résultats	2
3	Décodeur	3
3.1	Méthode	3
3.2	Résultat	3
4	Dropout en présence du décodeur	3
4.1	Méthode	3
4.2	Résultat	3
5	Ablation study	4
5.1	Utilisation d'un décodeur	4
5.2	Utilisation d'un VGAE	4
5.3	Conclusion	4
6	Reconstruction de graphe	5
6.1	Méthode	5
6.2	Résultats	5
6.3	Résultats avec connaissance du degré	6
7	Annexes	7
7.1	Performance	7
7.2	Reconstruction de graphe	10

Abstract

1 Méthode

Dans le cadre de ce projet, nous avons décidé de développer notre propre décodeur pour la prédiction de liens. Ce décodeur est composé de deux couches linéaires séparées par une fonction d'activation ReLu. Pour évaluer son efficacité, nous avons mis en place deux modèles :

- Un VGAE (Variational Graph Autoencoder) avec un encodeur composé d'une couche GCNConv suivie de deux couches GCNConv pour encoder la moyenne et l'écart type dans l'espace latent.
- Un GAE (Graph Autoencoder) composé d'une seule couche GCNConv.

Pour la préparation des données nous utilisons la méthode suivante :

```
torch_geometric.transforms.RandomLinkSplit(is_undirected=True, split_labels=True, num_val=0)
```

Cette méthode nous permet de générer des jeux d'entraînement et de test de liens existants dans les données d'origine (liens positifs) et non existants dans les données d'origine (liens négatifs). Elle nous permet aussi de définir que ces liens sont non orientés et que donc pour chaque paire d'indices formant le lien, le premier indice est inférieur au deuxième. Nous avons opté pour l'utilisation de l'optimiseur Adam avec un taux d'apprentissage de 0,01 et un terme de régularisation (`weight_decay`) de $5e^{-4}$. Tous les modèles présents auront une taille de sortie de l'espace latent de 32.

Dans l'ensemble de nos expérimentations, nous fournissons aux modèles de prédiction des données comprenant la latitude, la longitude et le pays (converti en un code numérique). En plus de cela, nous avons décidé d'explorer l'ajout de l'information sur le degré de chaque noeud et de comparer les résultats. Les degrés des noeuds sont déterminés en utilisant notre connaissance du graphe réel. Cependant, lorsque nous envisageons d'intégrer un nouvel aéroport à notre base de données, il devient envisageable d'estimer le degré potentiel du noeud associé à cet aéroport en se basant sur les flux d'arrivées et de départs des avions de cet aéroport. Ainsi, la réalisation de ce test nous semble pertinente et justifiée.

2 Dropout

2.1 Méthode

Nous avons essayé d'améliorer le modèle VGAE en y incorporant une couche de dropout entre la première et la deuxième couche. Nous avons laissé la valeur de dropout à la valeur par défaut soit 0.5. Les modèles ont été entraînés sur 2000 epochs. Ce processus a été répété 60 fois pour obtenir des statistiques sur l'apprentissage. Le processus a été réalisé à la fois dans la situation où le degré est inconnu et dans la situation où il est connu

2.2 Résultats

Les résultats au bout des 2000 epochs ont été compilé dans le tableau 1. L'évolution au cours de l'apprentissage de l'AUC et la précision moyenne ont, elles, été représenté sur la figure 1. Les résultats montrent sans équivoque que la présence de cette couche réduit énormément les performances du modèles sans réduire de manière suffisante l'écart-type et cela dans les deux situations. En effet, en regardant les courbes de la figure 1 on remarque que seul les instances les plus performantes de VGAE avec du dropout arrive à dépasser la médiane des instances de VGAE sans dropout. Pour la suite des tentatives on ne conservera donc pas le dropout.

modèle	Avec degré			Sans degré		
	AUC	AP	temps	AUC	AP	temps
VGAE	.823(.015)	75.6%(2.2%)	107.8s(4.9s)	.830(.023)	75.5%(2.8%)	110.9s(6.2s)
VGAE avec dropout	.785(.017)	70.6%(1.9%)	109.3s(5.3s)	.781(.021)	70.0%(2.6%)	112.0s(9.345)

TABLE 1 – Résultats de l'utilisation du dropout.

Dans chaque case est indiquée la moyenne et l'écart-type au format : moyenne(écart-type)

3 Décodeur

3.1 Méthode

Pour améliorer le VGAE nous avons décidé d'entraîner en plus de l'encodeur un décodeur. Le décodeur que nous avons entraîné est constitué de deux couches linéaires. La fonction d'activation entre les deux couches est ReLu. En sortie du décodeur la fonction d'activation est une sigmoïde. Les modèles ont été entraînés sur 2000 epochs. Ce processus a été répété 60 fois pour obtenir des statistiques sur l'apprentissage. Le processus a été réalisé à la fois dans la situation où le degré est inconnu et dans la situation où il est connu.

3.2 Résultat

Les résultats au bout des 2000 epochs ont été compilés dans le tableau 2. L'évolution au cours de l'apprentissage de l'AUC et la précision moyenne ont, elles, été représentées sur la figure 2. Les résultats montrent que l'utilisation du décodeur augmente très fortement les performances au détriment d'un écart type plus important. Sur la figure 2 on voit que l'augmentation d'écart-type est due à une petite portion des instances ayant une chute extrêmement forte (passant en dessous des instances sans décodeur).

modèle	Avec degré			Sans degré		
	AUC	AP	temps	AUC	AP	temps
VGAE avec décodeur	.960(.047)	95.8%(5.2%)	118.4s(6.4s)	.955(.061)	94.9%(7.0%)	122.8s(7.2s)
VGAE	.823(.015)	75.6%(2.2%)	107.8s(4.9s)	.830(.023)	75.5%(2.8%)	110.9s(6.2s)

TABLE 2 – Résultats de l'utilisation d'un décodeur.
Dans chaque case est indiquée la moyenne et l'écart-type au format : moyenne(ecart-type)

4 Dropout en présence du décodeur

4.1 Méthode

On réutilise la couche de dropout introduite dans la partie 2 et on la teste sur le vgae avec décodeur. En particulier on souhaite voir si son utilisation ne permettrait pas de réduire l'écart-type important introduit par le décodeur sans pour autant perdre trop en performance. Les modèles ont été entraînés sur 2000 epochs. Ce processus a été répété 60 fois pour obtenir des statistiques sur l'apprentissage. Le processus a été réalisé à la fois dans la situation où le degré est inconnu et dans la situation où il est connu.

4.2 Résultat

Les résultats au bout des 2000 epochs ont été compilés dans le tableau 2. L'évolution au cours de l'apprentissage de l'écart type de l'AUC et de la précision moyenne ont, elles, été représentées sur la figure 2. On remarque que l'écart-type est fortement réduit grâce aux dropout (divisé par 2 dans la situation où le degré est connu et par 10 lorsque celui-ci n'est pas). En plus de cela on peut remarquer que la moyenne elle-même augmente en effet les performances hautes sont peu impactées et les performances basses ont été fortement augmentées.

modèle	Avec degré			Sans degré		
	AUC	AP	temps	AUC	AP	temps
VGAE avec décodeur	.960(.047)	95.8%(5.2%)	118.4s(6.4s)	.955(.061)	94.9%(7.0%)	122.8s(7.2s)
VGAE avec décodeur et dropout	.962(.028)	96.0%(3.4%)	120.0s(6.6s)	.959(.006)	95.2%(0.7%)	122.8s(8.5s)

TABLE 3 – Résultats de l'utilisation du dropout en présence d'un décodeur.
Dans chaque case est indiquée la moyenne et l'écart-type au format : moyenne(écart-type)

5 Ablation study

Dans cette section on va vérifier que chacune des spécificités de notre modèle est bien nécessaire. La première spécificité est l'utilisation du dropout que l'on a justifié à la section précédente. La deuxième spécificité est l'utilisation d'un décodeur différent de celui par défaut. La dernière spécificité est l'utilisation de VGAE.

5.1 Utilisation d'un décodeur

Nous avons déjà vu que le décodeur avait un effet positif. Cependant, nous n'avons pas testé son efficacité en présence de dropout. Cependant, il n'y a pas besoin de refaire des calculs en effet. Avec le décodeur le dropout est meilleure et de plus sans le décodeur le dropout est moins bien. Ainsi :

$$\text{VGAE avec dropout} < \text{VGAE} < \text{VGAE avec décodeur} < \text{VGAE avec dropout et décodeur}$$

Donc le décodeur reste efficace en présence de dropout il est même encore plus efficace en présence de dropout.

5.2 Utilisation d'un VGAE

Nous allons maintenant vérifier la pertinence de l'utilisation d'un VGAE en le comparant avec un simple GAE. Les modèles ont été entraînés sur 2000 epochs. Ce processus a été répété 60 fois pour obtenir des statistiques sur l'apprentissage. Le processus a été réalisé à la fois dans la situation où le degré est inconnu et dans la situation où il est connu.

Les résultats ont été compilés dans le tableau 4. On remarque que dans le cas où le degré est connu Un simple GAE va avoir de meilleure performance que le VGAE. De plus le GAE mais 2 fois moins de temps pour réaliser le même nombre d'epochs. En revanche dans le cas plus compliqué où le degré n'est pas connu, le VGAE produit de meilleures performances au détriment d'un plus long temps de calcul et d'un plus grand écart-type. Sur la figure 4 on voit que VGAE atteint de meilleures performances dans la situation où le degré n'est pas connu, mais que certaines instances ont de très mauvaises performances faisant baisser les performances moyennes.

modèle	Avec Degré			Sans Degré		
	AUC	AP	temps	AUC	AP	temps
GAE avec décodeur	.968(.007)	96.9%(0.6%)	64.5s(3.5s)	.946(.004)	93.5%(0.5%)	67.1s(3.8s)
VGAE avec décodeur et dropout	.962(.028)	96.0%(3.4%)	120.0s(6.6s)	.959(.006)	95.2%(0.7%)	122.8s(8.5s)

TABLE 4 – Résultats de l'utilisation d'un GAE au lieu du VGAE.
Dans chaque case est indiquée la moyenne et l'écart-type au format : moyenne(écart-type)

5.3 Conclusion

Ainsi on peut voir que le décodeur et l'utilisation de dropout apporte réellement quelque chose à notre modèle. Cependant, dans la situation où le degré des noeuds est connu alors un simple GAE avec le décodeur

permet d'avoir de meilleur résultat avec une meilleure stabilité qu'avec un VGAE. En revanche dans le cas où le degré est inconnu le VGAE reste plus performant.

6 Reconstruction de graphe

6.1 Méthode

Nous avons choisi de tester nos modèles d'une manière plus visuelle, en cherchant à reconstruire un graphique représentant les liaisons aériennes entre les différents aéroports de notre ensemble de données. Pour ce faire, nous avons suivi une préparation des données similaire à celle décrite dans la section précédente. Nous avons ensuite entraîné nos modèles en utilisant une base de liens positifs pour l'entraînement. Par la suite, nous avons généré des graphiques en encodant les mêmes liens positifs et en décodant un ensemble de liens provenant de l'ensemble de données d'origine, ainsi qu'un nombre équivalent de liens négatifs. Nous avons ensuite enregistré plusieurs métriques, notamment le nombre de liens correctement prédits, le nombre de liens faussement prédits (qui n'existent pas dans nos données), le nombre de liens manqués (qui existent dans nos données, mais ont été rejetés par le modèle), et le nombre total de liens rejetés. Nous avons utilisé 4 modèles :

- Un GAE avec un encodeur formé d'une seule couche de convolution.
- Un VGAE sans notre décodeur.
- Un VGAE avec notre décodeur.
- Un même GAE mais avec notre décodeur.

Chaque modèle a été testé sur un total de 27 093 liens possibles, comprenant 13 547 liens positifs et 13 546 liens négatifs. L'objectif était de couper ce jeu de liens en deux en conservant les liens les plus plausibles pour le modèle. Pour ce faire, chaque modèle attribue un score entre 0 et 1 à chacun des 27 093 liens. Ensuite, à l'aide de la méthode `torch.quantile(z, 0.5)`, nous déterminons un seuil de score pour ne conserver que les $13\ 546 \pm 1$ liens ayant un score supérieur au seuil. Ce sont ces liens qui seront considérés comme prédits positivement par le modèle. Dans nos résultats, les pourcentages associés aux liens corrects et faux sont basés sur le nombre de liens considérés positifs par le modèle (Corrects + Faux) et le pourcentage de liens manqués par rapport au total de liens positifs réels (13 547). La prédiction est faite sur les données de **latitude**, **longitude**, et **pays** de chaque noeud.

Les liens en noir sont les liens correctement prédits.

Les liens en rouge sont les liens faussement prédits.

Les liens en vert sont les liens manqués.

6.2 Résultats

Modèle	Min	Max	Moyenne	Écart type	Seuil
GAE	0	1	0.75929	0.42746	1
VGAE	0	1	0.71422	0.45044	1
GAE avec décodeur	0	0.99980	0.49157	0.39727	0.54739
VGAE avec décodeur	0	1	0.56486	0.42113	0.75337

TABLE 5 – Statistiques des scores établies par les modèles

Modèle	Corrects		Faux		Manqués		Rejetés
GAE	13298	64.7%	7257	35.3%	249	1.84%	6538
VGAE	13381	71.0%	5473	29.0%	166	1.23%	8239
GAE avec décodeur	11804	87.1%	1743	12.9%	1743	12.9%	13546
VGAE avec décodeur	12199	90.0%	1348	10%	1348	10%	13546

TABLE 6 – Résultats de prédictions. voir figure 5 6 7 8

Il est notable que les modèles sans décodeur ont manqué beaucoup moins de liens par rapport aux modèles avec décodeur. Cependant, cette amélioration s'accompagne d'une plus grande acceptation de faux liens qui auraient dû être rejetés. Cette tendance s'explique en partie par la manière dont ces modèles attribuent des scores, sans faire de distinction nette entre les liens potentiellement positifs et les liens considérés comme certainement positifs.

6.3 Résultats avec connaissance du degré

Lors de nos tests, nous avons exploré l'incorporation des degrés de chaque nœud pour la prédiction de liens. Nous avons alors cherché à tester de manière plus visuelle nos modèles.

Modèle	Min	Max	Moyenne	Écart type	Seuil
GAE	0	1	0.73815	0.43963	1
VGAE	0	1	0.75625	0.41074	1
GAE avec décodeur	0	1	0.49266	0.45323	0.5414
VGAE avec décodeur	0	1	0.51289	0.43108	0.5917

TABLE 7 – Statistiques des scores établies par les modèles en tenant compte du degré des nœuds

Modèle	Corrects		Faux		Manqués		Rejetés
GAE	13239	66.2%	6745	33.8%	308	2.27%	6797
VGAE	11665	74.5%	3995	25.5%	1882	13.9%	11427
GAE avec décodeur	12044	88.9%	1501	11.1%	1502	11.1%	12044
VGAE avec décodeur	12535	92.5%	1011	7.5%	1012	7.5%	13547

TABLE 8 – Résultats de prédictions en tenant compte du degré des nœuds

Nous pouvons constater que tous les modèles obtiennent de meilleures performances lorsque le degré des nœuds est pris en compte. Notamment, dans ce contexte, le modèle GAE avec décodeur surpassé en performance le modèle VGAE avec décodeur.

7 Annexes

Ce situe ici les graphiques qui nous ont paru les plus pertinents pour la discussion des résultats. Cependant, d'autres graphiques ont été réalisés et peuvent être retrouvé dans le dossier graphique.

7.1 Performance

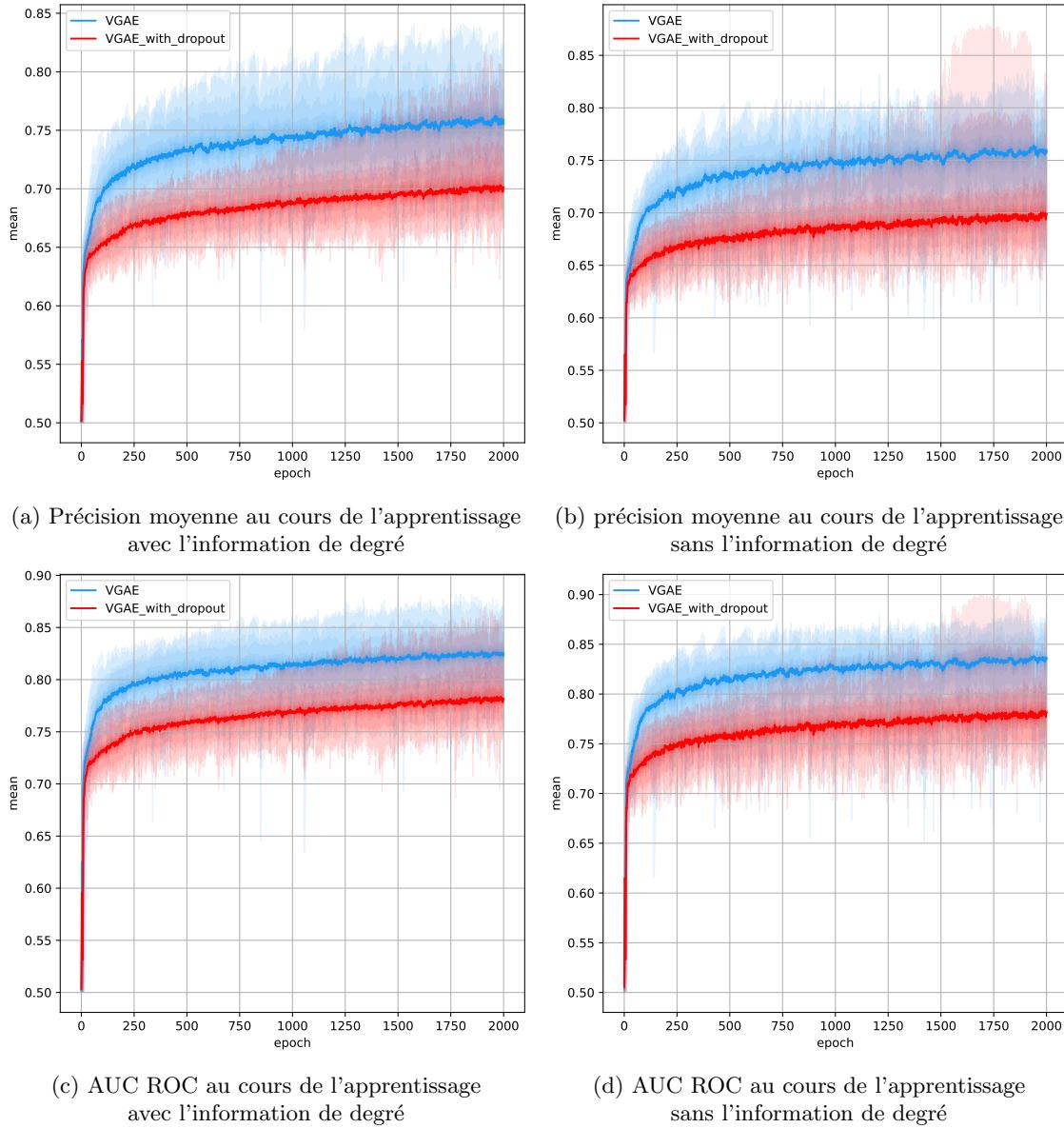


FIGURE 1 – Évolution de l'AUC et de la précision moyenne au cours de l'apprentissage dans les deux cas (degré connu ou inconnu) entre avec et sans dropout

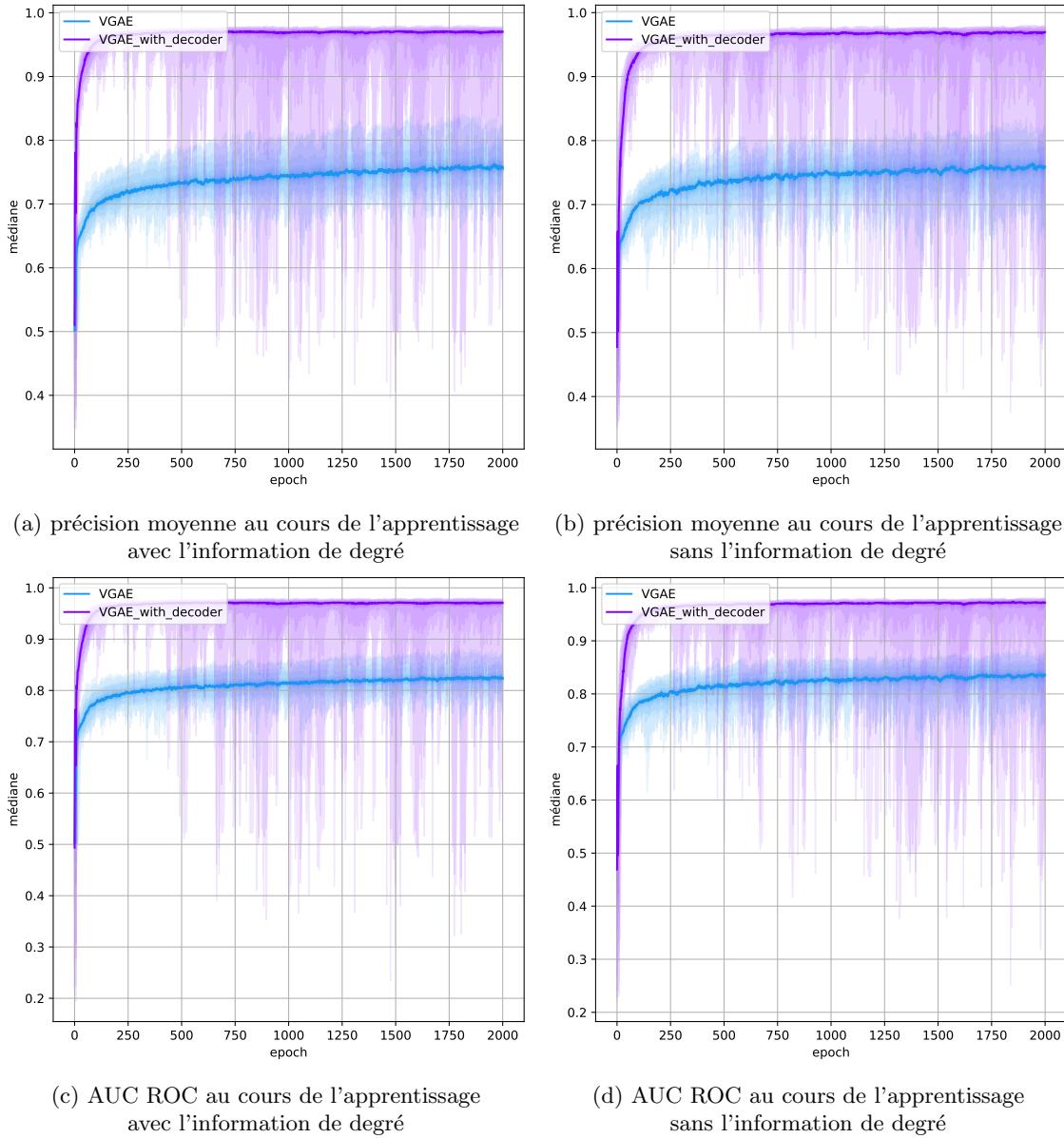
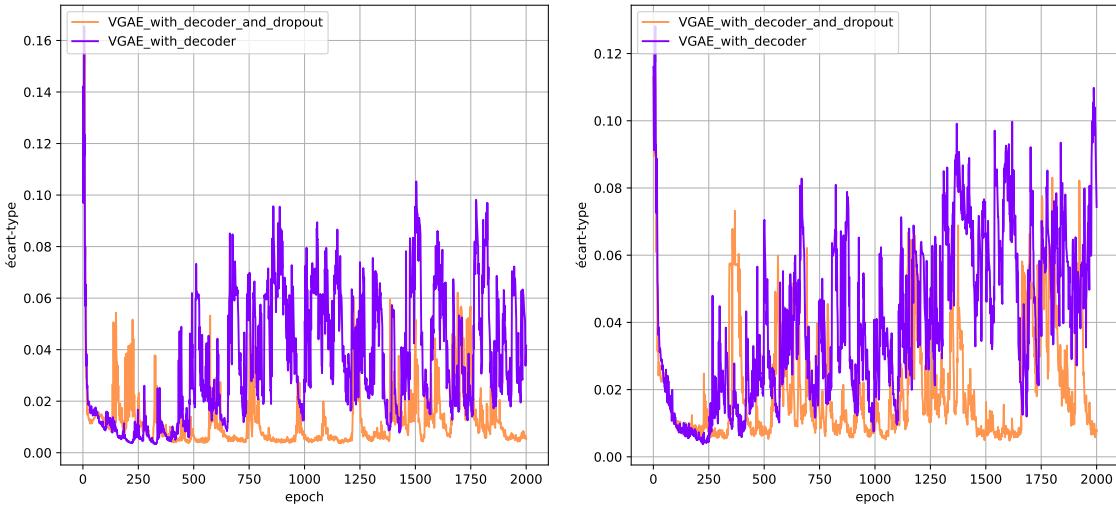
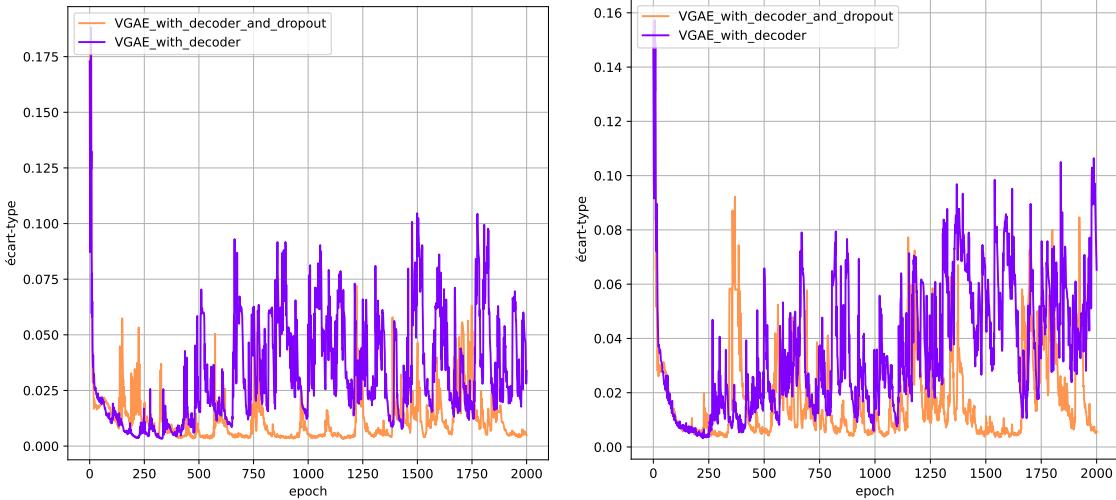


FIGURE 2 – Évolution de l'AUC et de la précision moyenne au cours de l'apprentissage dans les deux cas (degré connu ou inconnu) entre VGAE simple et VGAE avec décodeur



(a) précision moyenne au cours de l'apprentissage avec l'information de degré

(b) précision moyenne au cours de l'apprentissage sans l'information de degré



(c) AUC ROC au cours de l'apprentissage avec l'information de degré

(d) AUC ROC au cours de l'apprentissage sans l'information de degré

FIGURE 3 – Évolution de l'AUC et de la précision moyenne au cours de l'apprentissage dans les deux cas (degré connu ou inconnu) entre avec et sans dropout en présence du décodeur

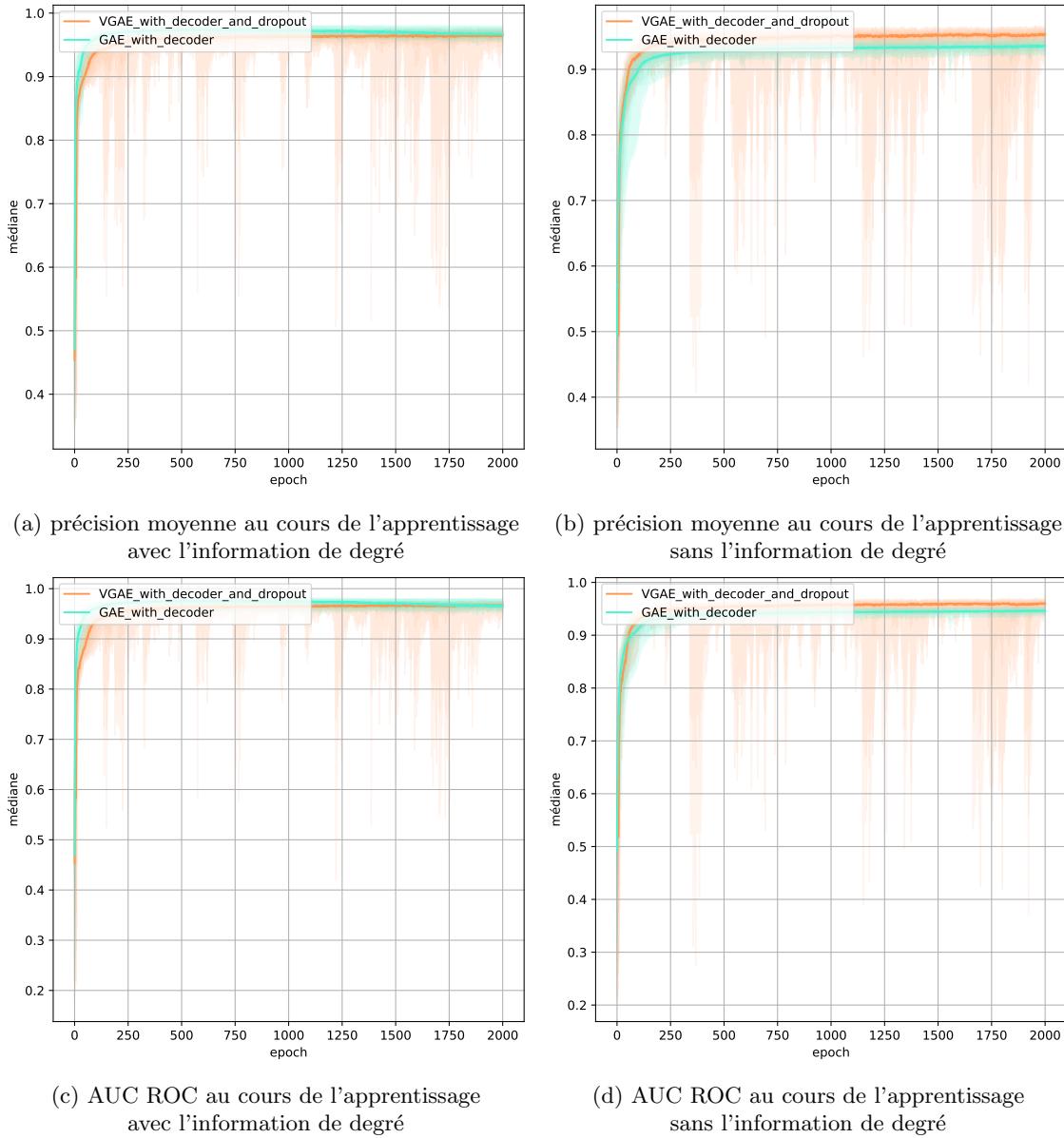


FIGURE 4 – Évolution de l'AUC et de la précision moyenne au cours de l'apprentissage dans les deux cas (degré connu ou inconnu) entre VGAE et GAE

7.2 Reconstruction de graphe

Les liens en noir sont les liens correctement prédis.
 Les liens en rouge sont les liens faussement prédis.
 Les liens en vert sont les liens manqués.

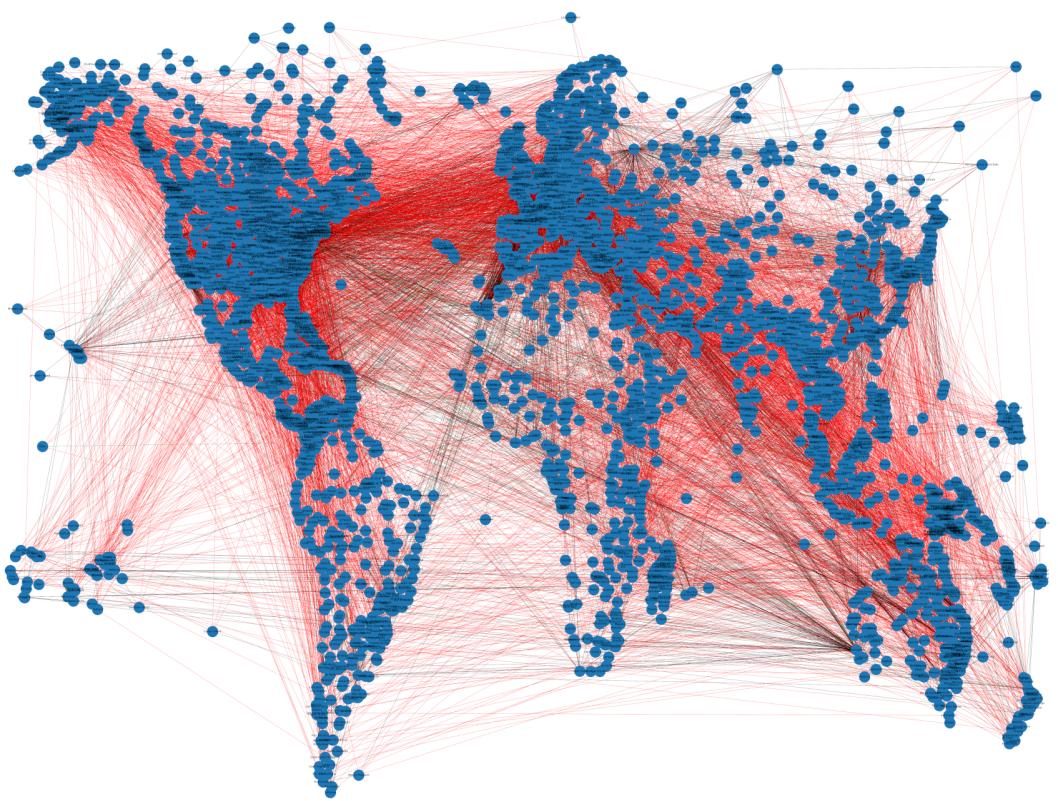


FIGURE 5 – Reconstruction du graphe avec le GAE



FIGURE 6 – Reconstruction du graphe avec le GAE muni de notre décodeur

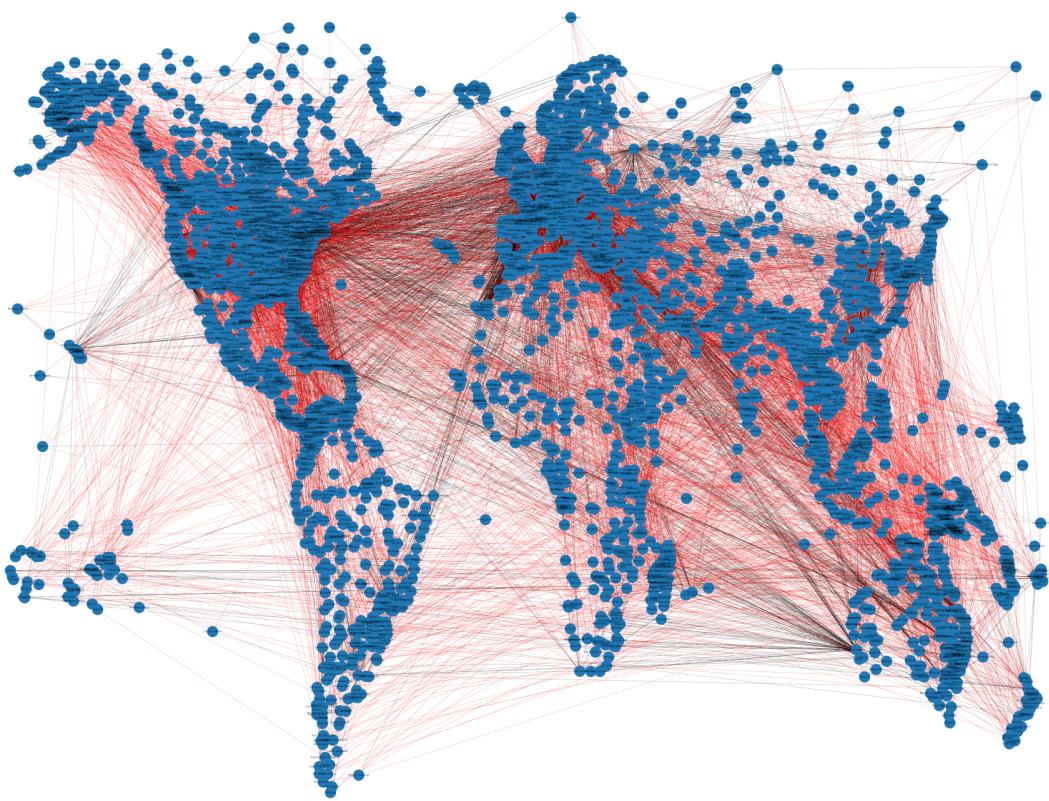


FIGURE 7 – Reconstruction du graphe avec le VGAE



FIGURE 8 – Reconstruction du graphe avec le VGAE muni de notre décodeur