徽标, 公司名称

描述已自动生成

Indoor Localization in IoT Networks Based on Graph Neural Networks

# Résumé

# Table des matières

[Résumé 1](#_Toc182866893)

[Introduction 3](#_Toc182866895)

[1 La localisation indoor 3](#_Toc182866896)

[1.1 Contexte 3](#_Toc182866897)

[1.2 L’Internet des Objets (IoT) 4](#_Toc182866898)

[1.3 Réseaux Neuronaux Profonds（DNN） 5](#_Toc182866899)

[1.4 Réseaux Neuronaux Graphiques（GNN） 6](#_Toc182866900)

[1.5 Base de Données : UJIIndoorLoc 7](#_Toc182866901)

[2 Méthode de Localisation Basée sur le DNN 7](#_Toc182866902)

[2.1 Fonctions Clés 8](#_Toc182866903)

[2.2 Structure du DNN 10](#_Toc182866904)

[2.3 Paramètres du Modèle 13](#_Toc182866905)

[2.4 Analyse des Paramètres et Résultats 14](#_Toc182866906)

[3 Méthode de Localisation Basée sur le GNN 16](#_Toc182866907)

[3.1 Fonctions du Modèle 16](#_Toc182866908)

[3.2 Structure du GNN 16](#_Toc182866909)

[3.3 Paramètres du Modèle 16](#_Toc182866910)

[3.4 Processus de Formation et Test 16](#_Toc182866911)

[3.5 Analyse des Résultats 16](#_Toc182866912)

[4 Comparaison entre les Méthodes DNN et GNN 16](#_Toc182866913)

[4.1 Comparaison de Performance 16](#_Toc182866914)

[4.2 Analyse des Avantages et Inconvénients 16](#_Toc182866915)

[4.3 Conclusion et Perspectives 16](#_Toc182866916)

[Références 17](#_Toc182866917)

# Introduction

## La localisation indoor

### Contexte

Avec le développement rapide de l'Internet des Objets (IoT), il occupe une place de plus en plus importante dans notre vie quotidienne. Dans les environnements intérieurs, les besoins en technologies de localisation augmentent, avec des applications variées allant des bâtiments intelligents aux centres commerciaux, en passant par les hôpitaux, etc. Cependant, les systèmes de localisation intérieure actuels présentent encore des lacunes en termes de précision et de fiabilité, ce qui limite leur capacité à répondre à la demande croissante en matière de navigation et de services basés sur la localisation (LBS).

Les principaux défis de la localisation intérieure incluent les aspects suivants :

* **Traitement de données multidimensionnelles** : par exemple, des signaux RSSI (Indicateur de puissance du signal reçu) comportant plus de dimensions. Ces données sont complexes, leur traitement représente un défi important.
* **Complexité des caractéristiques des coordonnées géographiques** : notamment la longitude, la latitude et l'étage, qui varient souvent de manière significative selon l'environnement et nécessitent une modélisation précise.
* **Optimisation de la précision des prédictions** : il est essentiel d'améliorer la sensibilité et l'exactitude des prédictions, en particulier pour les variations subtiles des données de localisation, afin de répondre aux exigences des scénarios d'utilisation réels.

Pour relever ces défis, ce projet vise à explorer l'efficacité de deux approches d'apprentissage profond dans le cadre de la localisation intérieure :

1. **Les réseaux de neurones profonds (DNN)** : capables de traiter des données complexes et multidimensionnelles, ces réseaux permettent d'extraire efficacement des caractéristiques à partir de larges ensembles de données.
2. **Les réseaux de neurones graphiques (GNN)** : en construisant une structure en graphe, ces réseaux capturent les relations spatiales entre les points d'accès Wi-Fi, améliorant ainsi la capacité du modèle à traiter des problèmes de localisation.

Ce projet se base sur la base de données UJIIndoorLoc, largement utilisée dans la recherche sur la localisation intérieure. En comparant les performances des DNN et des GNN pour la localisation intérieure, ce projet a pour objectif de discuter des problèmes liés à la précision de la localisation intérieure et de fournir des bases solides pour l'amélioration des services de localisation.

### L’Internet des Objets (IoT)

L'Internet des Objets (IoT) est considéré comme une avancée de l'Internet, dont l'objectif principal est de connecter le monde réel avec les technologies numériques. Grâce à l'IoT, les objets physiques peuvent être interconnectés via des réseaux de communication, permettant ainsi la génération et le partage de données. Le développement rapide de cette technologie a apporté de nombreuses opportunités dans divers domaines et a également eu un impact significatif sur les systèmes de localisation en intérieur.

Dans le domaine de la localisation en intérieur, l'IoT offre un soutien considérable. Par exemple, en déployant des capteurs et des équipements, l'IoT permet de capter en temps réel les environnements complexes à l'intérieur des bâtiments et de traiter des données multidimensionnelles.

Cependant, l'intégration de l'IoT dans la localisation en intérieur rencontre également certains défis, notamment les problèmes de confidentialité des données, la complexité de l'intégration des systèmes et les coûts élevés liés aux équipements et à leur maintenance.

1. **L'IoT repose sur trois composantes principales** [1]**:**

* **Objets physiques** : les équipements ou capteurs qui collectent et transmettent des données.
* **Réseaux de communication** : qui assurent l'interconnexion entre les équipements pour une transmission en temps réel des données.
* **Réseaux de traitement des données** : qui analysent et modélisent les données collectées pour en extraire des informations pertinentes.

1. **Avantages de l'IoT dans la localisation en intérieur** [1]**:**

* **Localisation précise** : Les besoins en localisation se divisent en position absolue (comme les coordonnées géographiques) et position relative (comme la position par rapport à des objets spécifiques dans l'environnement). Les technologies les plus récentes permettent d'atteindre une précision de localisation inférieure à 1 mètre, ce qui est essentiel pour des tâches comme la navigation ou le suivi dans des environnements complexes.
* **Réactivité en temps réel** : L'IoT peut capter les changements de l'environnement en temps réel, soutenant les scénarios dynamiques.
* **Applications étendues** : L'IoT s'applique à divers domaines tels que la santé, les bâtiments intelligents, la surveillance de la sécurité et la maintenance industrielle.

1. **Inconvénients de l'IoT dans la localisation en intérieur :**

* **Problèmes de confidentialité des données** : L'IoT implique une collecte et une transmission constantes de données, ce qui peut exposer les utilisateurs à des risques liés à la vie privée.
* **Complexité des systèmes** : Le déploiement des équipements, la gestion des réseaux et la maintenance nécessitent un haut niveau d'expertise technique.
* **Coûts élevés** : Les coûts associés à l'achat, à l'installation et à l'exploitation des équipements IoT peuvent constituer une barrière pour certains projets ou organisations aux ressources limitées.

À l'avenir, avec le développement et l'amélioration constants des technologies IoT, leur intégration avec la localisation en intérieur permettra d'améliorer encore davantage la précision et la réactivité tout en ouvrant la voie à des applications dans des scénarios plus larges.

### Réseaux Neuronaux Profonds（DNN）

Les réseaux neuronaux profonds (Deep Neural Networks, DNN) sont utilisés pour traiter des données multidimensionnelles complexes, permettant ainsi une classification et des prédictions efficaces et précises[2]. DNN se distinguent par la présence de plusieurs couches cachées, capables d'appliquer des transformations non linéaires successives aux données d'entrée. Cette structure permet de progressivement extraire des caractéristiques complexes et d'apprendre des relations de haut niveau entre les données. Les DNN capturent les caractéristiques des données d'entrée, ce qui les rend adaptés à des tâches telles que la classification et la régression[3].

1. **Structure Théorique** [4]**:**

* **Couche d'entrée** (Input Layer) : Reçoit les données d'entrée.
* **Couches cachées** (Hidden Layers) : Constituées de plusieurs couches entièrement connectées, chaque couche contenant un certain nombre de neurones. Ces couches réalisent une combinaison pondérée des entrées suivie de l'application d'une fonction d'activation, permettant d'extraire et d'apprendre des caractéristiques.
* **Couche de sortie** (Output Layer) : Détermine la sortie.
* **Fonction de perte** (Loss Function) : Utilisée pour mesurer l'écart entre les prédictions du modèle et les valeurs réelles.

1. **Avantages** [3] **:**

* **Traitement efficace de données multidimensionnelles** : Les DNN sont capables de gérer des données complexes et à haute dimension.
* **Extraction de caractéristiques profondes** : Grâce aux transformations non linéaires des couches cachées, les DNN peuvent extraire des caractéristiques abstraites et de haut niveau à partir des données.
* **Forte capacité de généralisation** : Avec un support de données suffisant, les DNN offrent de bonnes performances dans des scénarios complexes.

### Réseaux Neuronaux Graphiques（GNN）

Les réseaux neuronaux graphiques (Graph Neural Networks, GNN) sont des architectures spécialisées en apprentissage profond conçues pour traiter des données représentées sous forme de graphes. Elles exploitent la structure inhérente des graphes, ce qui les rend particulièrement adaptées aux tâches où les relations entre entités sont primordiales.

1. **Architecture typique :**

* **Couches Permutation Équivariantes** (Permutation Equivariant Layers) : implémentées comme des couches de passage de messages (Message Passing).
* **Couches de Pooling Local** (Local Pooling Layers) : simplifie le graphe en réduisant sa taille (downsampling).
* **Couches de Pooling Global** (Global Pooling Layers) : fournit une représentation de taille fixe pour l'ensemble du graphe.

1. **Avantages des GNN**[5] **:**

* **Adaptation aux Environnements Intérieurs Complexes** : La méthode de convolution dynamique capture bien les relations locales dans les données RSSI, ce qui rend les GNN efficaces dans des environnements intérieurs complexes.
* **Performance stable avec peu de données** : Même avec peu de données d’entraînement, les GNN gardent une grande précision et restent fiables.
* **Applications** : Les GNN peuvent être utilisés pour la navigation, le suivi d’objets, la gestion de bâtiments et des services personnalisés.

### Base de Données : UJIIndoorLoc

En tant que première base de données publique et la plus grande dans le domaine de la localisation et navigation en intérieur, UJIIndoorLoc permet de comparer différents algorithmes de localisation en intérieur, se distingue par ses caractéristiques suivantes : couverture multi-bâtiments et multi-étages, données diversifiées, grand volume d'échantillons, accès ouvert, et une méthode basée sur l'empreinte WLAN (WLAN fingerprinting).[6]

## Méthode de Localisation Basée sur le DNN

Dans l’application de localisation en intérieur, l’architecture DNN intègre les données hiérarchiques des bâtiments, étages et positions spécifiques, offrant une performance de localisation stable et une grande extensibilité, adaptée aux environnements complexes multi-bâtiments et multi-étages[2].

La méthode DNN proposée dans ce rapport est testée sur la base de données UJIIndoorLoc. Elle traite les données RSSI (les premières colonnes de chaque fichier de données) pour prédire la longitude, la latitude et l'étage des différentes coordonnées dans un même bâtiment. Les résultats sont comparés avec les valeurs de référence (colonnes à de chaque fichier de données). Le modèle fournit également le nombre d'erreurs de prédiction d'étage ainsi que la distance d'erreur en mètres pour les coordonnées de longitude et de latitude.

### Fonctions Clés

#### Fonctions d’Activation

ELU (Exponential Linear Unit) et ReLU (Rectified Linear Unit) sont des fonctions d'activation couramment utilisées dans les réseaux de neurones. Elles introduisent de la non-linéarité, permettant au modèle d'apprendre des caractéristiques plus complexes.

Bien que ReLU et ELU produisent une sortie identique pour les valeurs d'entrée positives, elles diffèrent pour les valeurs négatives : ReLU fixe la sortie à pour , tandis qu'ELU génère des valeurs négatives. Cela rend ELU plus adaptée aux données d'entrée comprenant des valeurs négatives.

Dans les fichiers de données utilisés pour ce modèle, les valeurs RSSI sont toutes négatives. Par conséquent, la fonction d'activation ELU a été choisie pour mieux s'adapter à cette distribution, renforçant ainsi la stabilité et l'efficacité de l'apprentissage du modèle.

La fonction d'activation ELU est définie comme suit, où est une constante positive :

Cette capacité d'ELU à produire des valeurs négatives pour permet de limiter le problème de disparition du gradient et d'améliorer la convergence du modèle. De plus, la sortie d'ELU permet aux moyennes des sorties des couches cachées de se rapprocher de , accélérant ainsi la vitesse d'entraînement du réseau.

#### Fonction de Perte

MSE (Mean Squared Error) est une fonction de perte couramment utilisée pour les tâches de régression. Elle est définie par la formule suivante :

Ce projet implique la prédiction de la longitude et de la latitude. La MSE est utilisée pour quantifier la différence entre les valeurs prédites et les valeurs réelles.

En minimisant la MSE, le modèle est incité à réduire les erreurs importantes, ce qui améliore la précision des prédictions de la longitude, de la latitude.

#### Optimiseur

Adam est un optimiseur adaptatif basé sur les estimations du premier et du second moment. En ajustant dynamiquement le taux d'apprentissage, Adam permet d'accélérer la vitesse de convergence du modèle tout en réduisant les fluctuations lors de la descente de gradient.

Dans ce projet, l'optimiseur Adam a amélioré l'efficacité de l'entraînement et a réduit le temps de formation du modèle DNN.

#### Prétraitement des Données

1. **Normalisation**

MinMaxScaler est une méthode de prétraitement des données qui permet de mettre à l'échelle les caractéristiques dans l'intervalle , tout en conservant les proportions relatives des valeurs des caractéristiques. Sa formule de transformation est la suivante :

Dans ce projet, MinMaxScaler est utilisé pour normaliser les données RSSI, permettant au modèle de mieux gérer les caractéristiques de différentes échelles et de réduire l'impact des différences d'ordre de grandeur entre les caractéristiques sur l'apprentissage du modèle. De plus, elle est également appliquée aux données de longitude et de latitude, en traitant les coordonnées géographiques séparément, le modèle est en mesure de prédire des valeurs de longitude et de latitude plus précises.

1. **Dénormalisation**

La dénormalisation permet de ramener les valeurs normalisées de sortie du modèle à leur échelle d'origine, rendant ainsi les prédictions interprétables et pertinentes dans un contexte réel. La formule de cette opération est la suivante :

Dans ce projet, la dénormalisation est utilisée pour ramener les valeurs prédites de longitude et de latitude du modèle de l'intervalle normalisé à leurs coordonnées géographiques d'origine. Cela permet de comparer directement les prédictions du modèle aux valeurs réelles, ce qui permet d'évaluer la précision du modèle.

#### Technique de Régularisation

Le Dropout est une technique de régularisation qui consiste à supprimer aléatoirement certaines sorties de neurones (en les fixant à ) pendant l'entraînement, réduisant ainsi la dépendance entre les neurones, empêchant le surapprentissage et améliorant la capacité de généralisation du modèle. Cette technique permet au modèle d’utiliser différentes combinaisons de neurones à chaque itération, ce qui équivaut à entraîner plusieurs sous-modèles et renforce ainsi la robustesse et la stabilité des prédictions.

### Structure du DNN

Pour le traitement des données d’un même bâtiment (building ), un modèle de DNN a été conçu, intégrant dans sa structure principale trois réseaux de neurones de régression. Chaque réseau est indépendant et dédié spécifiquement au traitement et à la prédiction de trois variables spatiales distinctes : la longitude, la latitude et l’étage.

#### Modèle d’Encodage

Ce modèle comprend deux couches d'encodage et deux couches de décodage.

1. Les couches d'encodage sont constituées de deux couches entièrement connectées, contenant respectivementet neurones, avec la fonction d'activation ELU.
2. Les couches de décodage restaurent les données pour reconstruire les caractéristiques d'entrée.

Ce modèle est utilisé pour extraire des caractéristiques des données RSSI de haute dimension et pour compresser la représentation, ce qui permet de réduire le bruit et les informations redondantes pendant la phase d'encodage, améliorant ainsi la précision des prédictions de longitude, latitude et étage tout en réduisant la charge de calcul.

#### Réseau de Régression

1. **La longitude**

Ce réseau de régression est spécialement conçu pour prédire la longitude, en effectuant des calculs de régression basés sur les caractéristiques extraites par les couches d'encodage.

Il comprend deux couches entièrement connectées avec la fonction d'activation ELU, une couche Dropout, et une couche de sortie à un seul neurone pour fournir le résultat de la prédiction de la longitude.

1. **La latitude**

Ce réseau de régression est spécialement conçu pour prédire la latitude, en effectuant des calculs de régression basés sur les caractéristiques extraites par les couches d'encodage.

Il comprend deux couches entièrement connectées avec la fonction d'activation ELU, une couche Dropout, et une couche de sortie à un seul neurone pour fournir le résultat de la prédiction de la latitude.

1. **L’étage**

Ce réseau de régression est conçu pour prédire l'étage.

Il comprend deux couches entièrement connectées avec la fonction d'activation ELU, ainsi qu'une couche de sortie à un seul neurone utilisant une fonction d'activation linéaire pour prédire le numéro de l'étage.

#### Couche de Détection d'Erreur

La couche de détection d'erreur est utilisée pour évaluer la précision des prédictions du modèle, en calculant l'écart entre les valeurs prédites et les valeurs réelles de la longitude, de la latitude et de l'étage.

La fonction error utilise l'erreur quadratique moyenne (Root Mean Square Error, RMSE) pour mesurer les résultats de prédiction des trois paramètres. RMSE reflète efficacement l'écart moyen entre les valeurs prédites et les valeurs réelles, les grandes erreurs ayant un impact plus important sur le résultat final.

Pour que l'erreur d'étage ait un poids plus important dans l'erreur totale, un coefficient de pénalité floor\_penalty est ajouté à l'erreur d'étage, ce qui permet au modèle d'accorder plus d'importance à la précision des prédictions d'étage.

Finalement, la fonction error génère des résultats comme Average Error in meters, utilisés pour l'analyse et l'évaluation des erreurs dans les étapes ultérieures.

En calculant séparément les erreurs de longitude, de latitude et d'étage, et en utilisant un coefficient de pénalité pour ajuster le poids de l'erreur d'étage, la fonction error permet une évaluation plus complète de la performance du modèle en termes de localisation. Elle prend en compte la précision continue des coordonnées géographiques tout en garantissant l'exactitude des prédictions d'étage, améliorant ainsi la performance globale de localisation du modèle.

#### Justification des Choix de Conception

Dans ce projet, pour la localisation au sein d'un même bâtiment, les données de latitude varient dans une plage relativement restreinte, tandis que les données de longitude montrent des variations plus importantes, avec des écarts de valeurs considérables entre les deux. Si ces deux paramètres ne sont pas traités séparément, le modèle risque d'ignorer les variations plus faibles de la longitude lors de l'entraînement. Lorsque la latitude et la longitude sont traitées par un même modèle, la latitude est prédite comme une valeur fixe.

Par conséquent, ce projet a conçu des réseaux indépendants pour la longitude, la latitude et l'étage, afin de garantir que le modèle puisse pleinement apprendre les particularités de chaque paramètre et effectuer des prédictions précises.

### Paramètres du Modèle

Dans le processus d'entraînement du DNN, il existe cinq paramètres importants pouvant être changés :

1. **Dimension d’entrée (input)**

La dimension d'entrée est directement déterminée par la structure du jeu de données.

Dans ce projet, chaque ligne de données contient caractéristiques RSSI, ce qui définit la dimension d'entrée à afin de garantir que le modèle puisse recevoir toutes les informations de caractéristiques.

1. **Taux de Dropout (dropout)**

Le dropout contrôle la proportion de neurones désactivés aléatoirement. Augmenter le taux de dropout peut éviter le surapprentissage, tandis que le réduire permet de conserver davantage d'informations.

Dans ce projet, le taux de dropout est défini à .

1. **Époques (epochs)**

L’epochs déterminent le nombre d'itérations sur l'ensemble de données.

Un nombre élevé d'itérations permet au modèle d’apprendre plus en profondeur les caractéristiques des données ; si un surapprentissage est détecté, il est possible de réduire cette valeur.

1. **Taille de Lot (batch\_size)**

La taille des lots (batch size) détermine le nombre d'échantillons utilisés pour chaque mise à jour des poids.

Dans la limite de la mémoire disponible, il est possible d'augmenter la taille des lots pour accélérer l'entraînement ; une taille de lot plus petite est préférable pour un jeu de données plus réduit et peut aider à réduire le surapprentissage.

1. **Coefficient de pénalisation (floor\_penalty)**

Le floor\_penalty détermine le poids de l'erreur d'étage dans l'erreur totale. Un coefficient de pénalité plus élevé permet au modèle d'accorder plus d'importance à la précision des prédictions d'étage.

Dans ce projet, le floor\_penalty est défini à .

### Analyse des Paramètres et Résultats

Dans ce projet, seuls les deux paramètres epochs et batch\_size sont ajustés et testés.

Étant donné que le nombre de données pour la longitude et la latitude est toujours le même et que celles-ci sont plus complexes, les paramètres epochs pour la longitude et la latitude sont définis de manière égale (longitude\_epochs = latitude\_epochs), tandis que les floor\_epochs sont fixés à une valeur plus petite.

Dans les légendes des graphiques, les abréviations suivantes sont utilisées :

* Le : longitude epochs et latitude epochs
* Fe: floor\_epochs
* BS: batch\_size

Les résultats sont les suivants :

图表, 折线图

描述已自动生成

Figure 1 Résultats d'erreur moyenne pour les données d'entraînement

图表, 折线图

描述已自动生成

Figure 2 Résultats d'erreur moyenne pour les données de test

Les conclusions suivantes peuvent être tirées des graphiques :

1. **Effet des époques (Epochs, Le & Fe)**

De manière générale, une augmentation du nombre d'époques conduit à une diminution de l'erreur moyenne. Cela montre que le modèle est capable d'apprendre davantage les caractéristiques des données avec plus d'itérations.

Cependant, une trop grande augmentation peut entraîner un risque de surapprentissage, en particulier avec de grands lots.

1. **Impact de la taille des lots (Batch Size, BS)**

Dans la majorité des cas, des lots de petite taille () entraînent des erreurs moyennes plus faibles. Cela peut s'expliquer par le fait que de petits lots permettent une meilleure capture des variations subtiles dans les données, notamment dans des ensembles de données complexes.

En revanche, des lots plus grands () offrent une stabilité mais peuvent augmenter légèrement l'erreur moyenne.

1. **Analyse par étage et complexité des données**

Lorsque les données sont regroupées par étage (), les performances du modèle sont plus stables et les erreurs restent faibles.

Pour le bâtiment complet () où les données de différents étages sont mélangées, une augmentation de l'erreur est observée, probablement due à la complexité accrue des données. Cependant, l'erreur globale reste dans une plage acceptable (moins de mètres).

1. **Meilleure combinaison de paramètres identifiée**

Parmi toutes les combinaisons testées, le meilleur résultat est les paramètres : .

## Méthode de Localisation Basée sur le GNN

简介GNN在定位的用法、优势等信息。

### Fonctions du Modèle

### Structure du GNN

### Paramètres du Modèle

### Processus de Formation et Test

### Analyse des Résultats

## Comparaison entre les Méthodes DNN et GNN

### Comparaison de Performance

### Analyse des Avantages et Inconvénients

### Conclusion et Perspectives

# Références

1. Njima, Wafa. Méthodes de localisation de capteurs dans le contexte de l’Internet des Objets. Traitement des images [eess.IV]. Conservatoire national des arts et metiers – CNAM ; École supérieure des communications de Tunis (Tunisie), 2019, pp.22-23. NNT : 2019CNAM1264. URL : https://tel.archives-ouvertes.fr/tel-02484757
2. Kim, K.S., Lee, S., & Huang, K. (2018). A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting. Big Data Analytics, 3(4). https://doi.org/10.1186/s41044-018-0031-2
3. H. Asadollahi, N. Mokari, H. Saeedi, H. Yanikomeroglu, “Enhancing Indoor Localization Accuracy Using a Hybrid Deep Learning Algorithm: A DNN-CNN Approach,” 2023.
4. W. Njima, A. Bazzi, M. Chafii, “DNN-Based Indoor Localization Under Limited Dataset Using GANs and Semi-Supervised Learning,” IEEE Access, vol. 10, pp. 69896–69910, 2022. DOI : 10.1109/ACCESS.2022.3187837
5. Vishwakarma, R., Joshi, R.B., & Mishra, S. (2023). IndoorGNN: A Graph Neural Network Based Approach for Indoor Localization using WiFi RSSI.
6. J. Torres-Sospedra, R. Montoliu, A. Martínez-Usó, et al., "UJIIndoorLoc : A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems," IPIN 2014, 2014. DOI : 10.1109/IPIN.2014.7275492